AWS ホワイトペーパー

AWS Outposts の高可用性設計とアーキテクチャに関する考慮事項

AWS Outposts の高可用性設計とアーキテクチャに関する考慮事項: AWS ホワイトペーパー

Table of Contents

AWS Outposts の高可用性設計とアーキテクチャに関する考慮事項

出版日: 2021 年 8 月 12 日 (ドキュメント履歴)

このホワイトペーパーでは、IT マネージャーやシステムアーキテクトが、AWS Outposts により可用性の高いオンプレミスアプリケーション環境を構築するために適用できるアーキテクチャに関する考慮事項と推奨プラクティスについて説明します。

Well-Architected の実現状況の確認

AWS Well-Architected フレームワークは、クラウド内でのシステム構築に伴う意思決定の長所と短所を理解するのに役立ちます。このフレームワークの 6 つの柱により、信頼性、安全性、効率、費用対効果、持続可能性の高いシステムを設計および運用するための、アーキテクチャのベストプラクティスを確認できます。AWS Management Console で無料で提供されている AWS Well-Architected Tool を使用すると、柱ごとに一連の質問に答えることで、これらのベストプラクティスに照らしてワークロードを評価できます。

クラウドアーキテクチャに関する専門的なガイダンスやベストプラクティス (リファレンスアーキテクチャのデプロイ、図、ホワイトペーパー) については、AWS アーキテクチャセンターを参照してください。

序章

このホワイトペーパーは、AWS クラウドプラットフォームを使用してアプリケーションをデプロイ、移行、運用し、それらのアプリケーションを AWS Outposts ラック (AWS Outposts の 42U ラックフォームファクタ) によりオンプレミスで実行することを検討している IT マネージャーやシステムアーキテクトを対象としています。

AWS Outposts ラックを含む高可用性システムを構築するためのアーキテクチャパターン、アンチパターン、および推奨プラクティスを紹介します。AWS Outposts ラックのキャパシティを管理し、ネットワーキングとデータセンターファシリティサービスを使用して可用性の高い AWS Outposts ラックインフラストラクチャソリューションをセットアップする方法を学びます。

AWS Outposts ラックは、クラウドコンピューティング、ストレージ、ネットワーク機能の論理プールを提供するフルマネージドサービスです。Outposts ラックを使用すると、サポートされている AWS マネージドサービスをオンプレミス環境で使用できます。これには、Amazon Elastic

Compute Cloud (Amazon EC2)、Amazon Elastic Block Store (Amazon EBS)、 Amazon S3 on Outposts、Amazon Elastic Kubernetes Service (Amazon EKS)、Amazon Elastic Container Service (Amazon ECS)、Amazon Relational Database Service (Amazon RDS)、およびその他の Outposts 上の AWS のサービスが含まれます。Outposts のサービスは、AWS リージョンで使用されているのと同じ AWS Nitro システムで提供されます。

AWS Outposts ラックを活用することで、使い慣れた AWS のクラウドサービスとツールを使用して、可用性の高いオンプレミスアプリケーションを構築、管理、スケールできます。AWS Outposts は、オンプレミスシステムへの低レイテンシーアクセス、ローカルデータ処理、データレジデンシー、およびローカルシステムの相互依存関係を持つアプリケーションの移行を必要とするワークロードに最適です。

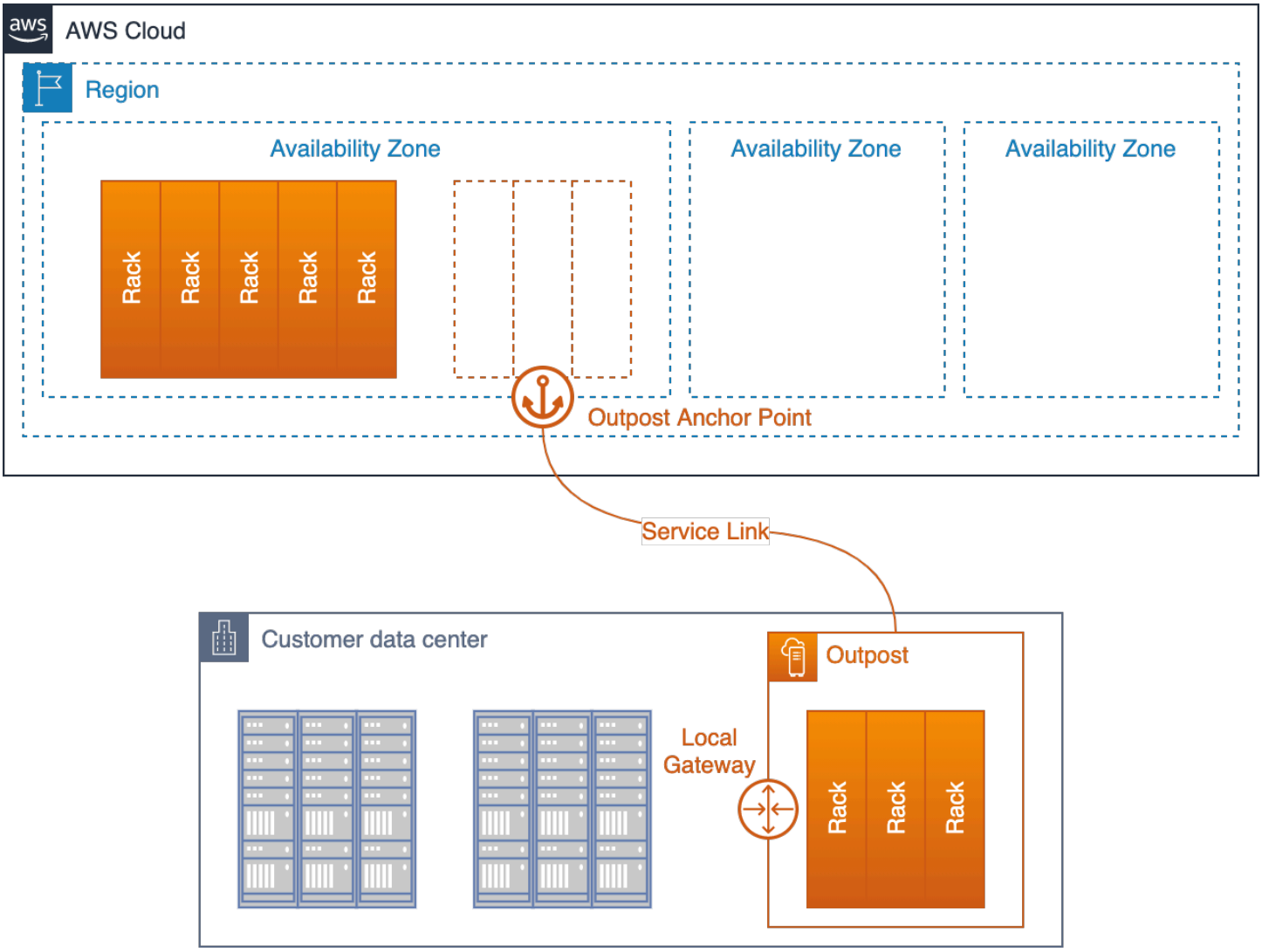
AWS のインフラストラクチャとサービスをオンプレミスの場所に拡張

AWS Outposts サービスでは、50 を超える国と地域のオンプレミス拠点に AWS インフラストラクチャとサービスを提供し、顧客は事実上すべてのデータセンター、コロケーションスペース、またはオンプレミス施設に同じ AWS インフラストラクチャ、AWS サービス、API、ツールをデプロイして、真に一貫したハイブリッドエクスペリエンスを実現できます。Outposts での設計方法を理解するには、AWS クラウドを構成するさまざまな階層を理解する必要があります。

AWS リージョン は世界の地理的領域です。AWS リージョン のそれぞれは、アベイラビリティーゾーン (AZ) に論理的にグループ化されたデータセンターの集まりです。AWS リージョン は、低レイテンシー、高スループット、冗長ネットワーク接続で接続されている、物理的に分離された複数の (少なくとも 2 つの) アベイラビリティーゾーンを提供します。各 AZ は 1 つ以上の物理データセンターで構成されています。

論理 Outpost (以下、Outpost) とは、物理的に接続された 1 つ以上の AWS Outposts ラックを 1 つのエンティティとして管理するデプロイです。Outpost は、AWS リージョン 内の AZ のプライベートエクステンションとして、いずれかのサイトに AWS コンピューティング容量とストレージ容量のプールを提供します。

おそらく最良の AWS Outposts の概念モデルは、AWS リージョン の AZ にあるデータセンターから 1 つまたは複数のラックを取り外し、独自のデータセンターまたはコロケーション施設にインストールすることを考えることです。AZ データセンターからご使用のデータセンターまでラックを展開します。次に、(非常に) 長いケーブルでラックを AZ データセンターのアンカーポイントに差し込み、ラックが引き続き AWS リージョン の一部として機能するようにします。また、それらをローカルネットワークに接続して、オンプレミスネットワークとそれらのラックで実行されているワークロードとの間を低レイテンシーで接続できるようにします。これにより、ワークロードをローカルに保ちながら、AWS クラウド の運用と API の一貫性が得られます。



顧客のデータセンターにデプロイされ、アンカー AZ と親リージョンに接続し直した Outpost

Outpost は、アンカーされている AZ の延長として機能します。AWS は、AWS リージョン の一部として AWS Outposts インフラストラクチャの運用、モニタリング、管理を行います。Outpost は、非常に長い物理ケーブルの代わりに、サービスリンクと呼ばれる暗号化された VPN トンネルのセットを経由して親リージョンに接続し直します。

サービスリンクは、Outpost の親リージョンのアベイラビリティーゾーン (AZ) にある一連のアンカーポイントで終了します。

コンテンツを保存する場所を選択します。コンテンツは、AWS リージョン または他の場所に複製およびバックアップできます。法律または政府機関の拘束力のある命令に従うために必要な場合を除き、お客様の同意なしに、お客様のコンテンツが選択した場所以外に移動またはコピーされることはありません。詳細については、AWS データプライバシーのよくある質問を参照してください。

これらのラックでデプロイするワークロードはローカルで実行されます。また、これらのラックで利用できるコンピューティングとストレージの容量には限りがあり、AWS リージョン のクラウドスケールのサービスの実行に対応することはできませんが、ラックにデプロイされたリソース (インスタンスとそのローカルストレージ) はローカルで実行され、管理プレーンは引き続き AWS リージョン で稼働するというメリットがあります。

Outpost にワークロードをデプロイするには、仮想プライベートクラウド (VPC) 環境にサブネットを追加し、サブネットの場所として Outpost を指定します。次に、AWS Management Console、CLI、API、CDK、またはコードとしてのインフラストラクチャ (IaC) ツールを使用してサポートされている AWS リソースをデプロイするときに、目的のサブネットを選択します。Outpost サブネットのインスタンスは、VPC のネットワーキングを通じて Outpost またはリージョン内の他のインスタンスと通信します。

Outpost のサービスリンクは、Outpost 管理トラフィックとカスタマー VPC トラフィック (Outpost 上のサブネットとリージョン内のサブネット間の VPC トラフィック) の両方を伝送します。

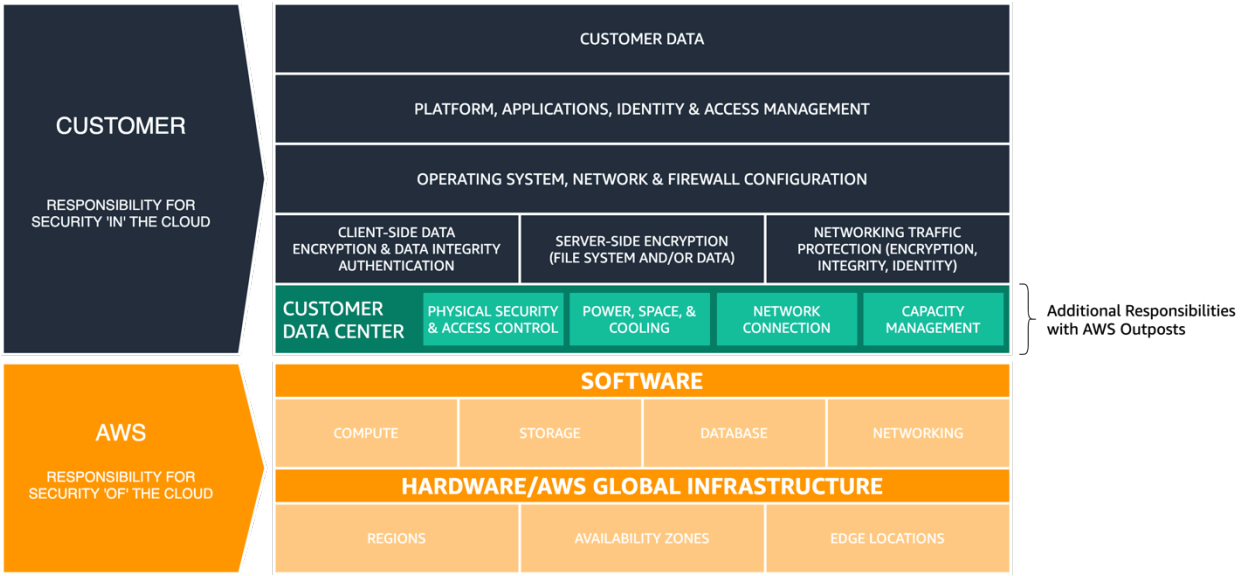
重要な用語:

- AWS Outposts — は、実質的にすべてのデータセンター、コロケーションスペース、またはオンプレミス施設に同じ AWS インフラストラクチャ、AWS サービス、API、およびツールを提供するフルマネージドサービスであり、真に一貫したハイブリッドエクスペリエンスを実現します。
- Outpost — は、物理的に接続された AWS Outposts ラックを 1 つの論理エンティティとして管理し、お客様のサイトに AWS コンピューティング、ストレージ、ネットワーキングのプールをデプロイした環境です。
- 親リージョン — Outpost のデプロイに必要な管理、コントロールプレーンサービス、およびリージョンレベルの AWS サービスを提供する AWS リージョン です。
- アンカーアベイラビリティーゾーン (アンカー AZ) — Outpost のアンカーポイントをホストする親リージョンのアベイラビリティーゾーン。Outpost はアンカー AZ の延長として機能します。アンカー AZ は、Outposts の注文時にお客様が選択します。アンカー AZ を選択すると、AWS Outposts サブスクリプションの期間中は変更できません。
- アンカーポイント — リモートでデプロイされた Outposts からの接続を受信するアンカー AZ 内のエンドポイント。
- サービスリンク — Outpost を親リージョンのアンカーアベイラビリティーゾーンに接続する暗号化された VPN トンネルのセット。
- ローカルゲートウェイ (LGW) — Outpost とオンプレミスネットワーク間の通信を可能にする論理的な相互接続仮想ルーター。

AWS Outposts の責任共有モデルを理解する

データセンターやコロケーション施設に AWS Outposts インフラストラクチャをデプロイするとき、AWS 責任共有モデルでは追加の責任を引き受けることになります。例えば、リージョンでは、AWS はさまざまな電源、冗長コアネットワーク、および耐障害性の高いワイドエリアネットワーク (WAN) 接続を提供して、1 つ以上のコンポーネントに障害が発生した場合でもサービスを確実に利用できるようにします。

Outposts では、Outposts 上で実行されるワークロードの可用性要件を満たすために、Outposts ラックに回復力のある電力とネットワーク接続を供給する必要があります。



AWS Outposts 用に更新された AWS 責任共有モデル

AWS Outposts では、データセンター環境の物理的セキュリティとアクセスコントロールはお客様の責任となります。Outpost を稼働させ続けるために十分な電力、スペース、冷却装置を用意し、Outpost をリージョンに接続するためのネットワーク接続を提供する必要があります。

Outpost の容量には限りがあり、AWS がサイトに設置するラックのサイズと数によって決定されるため、初期ワークロードを実行し、将来の増加に対応して、サーバーの障害やメンテナンスイベントを軽減するために追加容量を提供するために必要な EC2、EBS、S3 on Outposts の容量を決定する必要があります。

AWS は、AWS Outposts ラック内の電源、サーバー、ネットワーク機器を含む Outposts インフラストラクチャの可用性に責任を持ちます。また、AWS は、Outposts で実行される仮想化ハイパーバイザー、ストレージシステム、AWS サービスを管理します。

各 Outposts ラックの中央電源シェルフは、AC 電源から DC 電源に変換し、バスバーアーキテクチャを介してラック内のサーバーに電力を供給します。バスバーアーキテクチャでは、ラック内の電源の半分が故障しても、すべてのサーバーが中断することなく稼働し続けます。

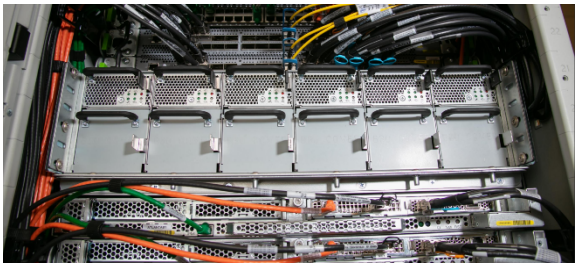


図 3 - AWS Outposts AC/DC 電源とバスバーの配電

Outposts ラック内およびラック間のネットワークスイッチとケーブルも完全に冗長です。ファイバーパッチパネルは、Outpost ラックとオンプレミスネットワークを接続し、顧客管理のデータセンター環境とマネージド AWS Outposts 環境の間の境界点として機能します。

リージョンでと同様に、AWS は Outposts で提供されるクラウドサービスに責任を持ち、お客様が Amazon RDS on Outposts などの上位のマネージドサービスを選択してデプロイすると、追加の責任を引き受けます。Outposts にデプロイするサービスを検討および選択する際には、個々のサービスの「AWS 責任共有モデル」ページと「よくある質問 (FAQ)」ページを確認する必要があります。これらのリソースには、お客様と AWS との責任分担に関する追加情報が記載されています。

障害モードの観点からの考え方

可用性の高いアプリケーションやシステムを設計する際には、障害が発生する可能性のあるコンポーネント、コンポーネントの障害がシステムとアプリケーションの RPO/RTO 目標に与える影響、およびコンポーネントの障害の影響を軽減または排除するために実装できるメカニズムを検討する必要があります。アプリケーションは 1 台のサーバー、1 台のラック、1 つのデータセンターで実行されますか? サーバー、ラック、またはデータセンターに一時的または永続的な障害が発生した場合はどうなりますか? ネットワークなどの重要なサブシステムやアプリケーション自体に障害が発生した場合はどうなりますか? これらは、障害モードです。

Outposts とアプリケーションのデプロイを計画する際には、このセクションの障害モードを考慮する必要があります。以下のセクションでは、これらの障害モードを軽減してアプリケーション環境の高可用性を向上させる方法について説明します。

障害モード 1: ネットワーク

Outpost のデプロイメントは、管理とモニタリングにおいて親リージョンへのレジリエントな接続が不可欠です。ネットワークの中断は、オペレーターのミス、機器の故障、サービスプロバイダーの停止など、さまざまな障害によって引き起こされる可能性があります。Outposts は、サイトで相互に接続された 1 つ以上のラックで構成されている場合がありますが、サービスリンクを介してリージョンと通信できない場合、接続が切断されていると見なされます。

冗長なネットワークパスは、切断イベントのリスクを軽減するのに役立ちます。接続解除イベントがワークロード操作に与える影響を把握するには、アプリケーションの依存関係とネットワークトラフィックをマッピングする必要があります。アプリケーションの可用性要件を満たすために、十分なネットワーク冗長性を計画してください。

切断イベントが発生しても、Outpost で実行されているインスタンスは引き続き実行され、Outpost ローカルゲートウェイ (LGW) を介してオンプレミスネットワークからアクセスできます。ローカルのワークロードやサービスは、リージョンのサービスに依存している場合、問題や失敗が発生する可能性があります。Outpost がリージョンから切断されている間は、変更リクエスト (Outpost でのインスタンスの起動や停止など)、コントロールプレーンオペレーション、サービステレメトリ (CloudWatch メトリクスなど) は失敗します。CloudWatch メトリクスは、短時間のネットワーク切断の間、Outpost 上でローカルにスプールされ、サービスリンク接続が再確立されたときに確認のためにリージョンに送信されます。

障害モード 2: インスタンス

Amazon EC2 インスタンスでは、実行中のサーバーに問題があったり、インスタンスにオペレーティングシステムやアプリケーションの障害が発生したりすると、問題や失敗が発生する可能性があります。アプリケーションでこれらのタイプの障害をどのように処理するかは、アプリケーションのアーキテクチャによって異なります。モノリシックアプリケーションは通常、リカバリのためにアプリケーションまたはシステム機能を使用しますが、モジュラー型のサービス指向またはマイクロサービスアーキテクチャでは、障害が発生したコンポーネントを交換してサービスの可用性を維持するのが一般的です。

Amazon EC2 Auto Scaling グループなどの自動メカニズムを使用して、失敗したインスタンスを新しいインスタンスに置き換えることができます。インスタンスの自動リカバリでは、残りのサーバーに十分な空き容量があり、サービスリンクがまだ接続されている限り、サーバー障害が原因で失敗したインスタンスを再起動できます。

障害モード 3: コンピューティング

サーバーでは、失敗や問題が発生する可能性があり、コンポーネントの障害や定期的なメンテナンス作業など、さまざまな理由で (一時的または永続的に) 運用を停止する必要がある場合があります。Outposts ラックのサービスがサーバーの障害を処理する方法はさまざまで、お客様が高可用性オプションをどのように設定するかによっても異なります。

N+M 可用性モデルをサポートするのに十分なコンピューティング性能を注文する必要があります。ここで、N は必要な性能、M はサーバー障害に対応するためにプロビジョニングされた予備性能です。

障害が発生したサーバーのハードウェア交換は、フルマネージド AWS Outposts ラックサービスの一環として提供されます。AWS は、Outpost デプロイ内のすべてのサーバーとネットワークデバイスの状態を積極的にモニタリングします。物理的なメンテナンスを行う必要がある場合、AWS は故障したコンポーネントを交換するためにサイトを訪問する時間をスケジュールします。予備の性能をプロビジョニングすることで、障害が発生したサーバーがサービスを停止して交換される間も、ホスト障害に対するワークロードの耐障害性を維持できます。

障害モード 4: ラックまたはデータセンター

ラックの障害は、ラックの電力が完全に失われることや、洪水や地震による冷却の喪失やデータセンターの物理的損傷などの環境の障害が原因で発生する可能性があります。データセンターの配電

アーキテクチャの欠陥やデータセンターの標準的な電源メンテナンス中のエラーは、1 つまたは複数のラック、さらにはデータセンター全体の電力が失われる原因になる可能性があります。

これらのシナリオは、同じキャンパスまたは大都市圏内の互いに独立した複数のデータセンターのフロアまたは場所にインフラストラクチャをデプロイすることで軽減できます。

AWS Outposts ラックでこのアプローチを採用する際には、アプリケーションの可用性を維持するために、アプリケーションを複数の異なる論理的な Outposts にわたって実行するように設計および分散する方法を慎重に検討する必要があります。

障害モード 5: AWS アベイラビリティーゾーンまたはリージョン

各 Outpost は、AWS リージョン 内の特定のアベイラビリティーゾーン (AZ) にアンカーされます。アンカー AZ または親リージョン内で障害が発生すると、Outpost の管理や変更可能性が失われ、Outpost とリージョンの間のネットワーク通信が中断される可能性があります。

ネットワーク障害と同様に、AZ またはリージョンの障害により、Outpost がリージョンから切断されることがあります。Outpost で実行されているインスタンスは引き続き実行され、Outpost ローカルゲートウェイ (LGW) を介してオンプレミスネットワークからアクセスできますが、前述のように、リージョンのサービスに依存している場合は問題や失敗が発生する可能性があります。

AWS AZ とリージョンの障害による影響を軽減するために、それぞれ異なる AZ またはリージョンにアンカーされた複数の Outposts をデプロイできます。次に、現在 AWS で設計およびデプロイに使用している同様のメカニズムとアーキテクチャパターンの多くを使用して、分散型のマルチ Outpost アウトポストデプロイモデルで動作するようにワークロードを設計できます。

AWS Outposts で実行されるサービスのコントロールプレーンは、アンカーされているリージョンに存在し、Amazon EC2 や Amazon EBS などのゾーンサービス、および Amazon RDS、Elastic Load Balancing、Amazon EKS などのリージョンサービスの両方に依存します。Outposts では、アプリケーションを静的安定性の概念でデプロイして、コントロールプレーンの障害に対する回復力を向上させることができます。

AWS Outposts ラックによる HA アプリケーションとインフラストラクチャソリューションの構築

AWS Outposts ラックを使用すると、使い慣れた AWS のクラウドサービスとツールを使用して、可用性の高いオンプレミスアプリケーションを構築、管理、スケールすることができます。クラウド HA アーキテクチャとアプローチは、現在データセンターで運用している従来のオンプレミス HA アーキテクチャとは全般的に異なることを理解することが重要です。

従来のオンプレミスの HA アプリケーションのデプロイでは、アプリケーションは仮想マシン (VM) にデプロイされます。複雑な IT システムとインフラストラクチャがデプロイおよび管理され、仮想マシンの実行と健全性を維持します。多くの場合、VM には特定の ID があり、各 VM はアプリケーションアーキテクチャ全体で重要な役割を果たす場合があります。

アーキテクチャ上の役割は VM の ID と密接に結びついています。システムアーキテクトは、IT インフラストラクチャ機能を活用して可用性の高い VM ランタイム環境を提供し、各 VM がコンピューティング容量、ストレージボリューム、およびネットワークサービスに確実にアクセスできるようにします。VM に障害が発生した場合、自動または手動の復旧プロセスを実行して、障害が発生した VM を (多くの場合、他のインフラストラクチャ上または完全に別のデータセンターで) 健全な状態に復元します。

クラウド HA アーキテクチャは異なるアプローチを採用しています。AWS クラウドサービスは、信頼性の高いコンピューティング、ストレージ、およびネットワーク機能を提供します。アプリケーションコンポーネントは EC2 インスタンス、コンテナ、サーバーレス機能、またはその他のマネージドサービスにデプロイされます。

インスタンスはアプリケーションコンポーネントをインスタンス化したもので、その役割を果たす多くのコンポーネントのうちの 1 つである可能性があります。アプリケーションコンポーネントは相互に、およびアプリケーションアーキテクチャ全体で果たす役割と疎結合しています。通常、インスタンスの個々の ID は重要ではありません。需要に応じてスケールアップまたはスケールダウンするために、追加のインスタンスを作成または破棄できます。障害が発生したインスタンスや異常なインスタンスは、単に新しい正常なインスタンスに置き換えられます。

AWS Outposts ラックは、AWS のコンピューティング、ストレージ、ネットワーキング、データベース、およびその他のクラウドサービスをオンプレミスの場所に拡張し、真に一貫したハイブリッドエクスペリエンスを実現するフルマネージドサービスです。Outposts ラックサービスを、従来のオンプレミスの HA メカニズムを備えた IT インフラストラクチャシステムの当座の代用と考えるべ

きではありません。従来のオンプレミスの HA アーキテクチャをサポートするために AWS サービスや Outposts を使用しようとするのはアンチパターンです。

AWS Outposts ラックで実行中のワークロードは、Amazon EC2 Auto Scaling (ワークロードの需要に合わせて水平にスケール)、EC2 ヘルスチェック (異常のあるインスタンスを検出して削除)、Application Load Balancer (受信ワークロードトラフィックをスケーリングまたは置換されたインスタンスにリダイレクト) などのクラウド HA メカニズムを使用します。アプリケーションをクラウドに移行する際は、移行先が AWS リージョン であるか AWS Outposts ラックであるかにかかわらず、HA アプリケーションのアーキテクチャを更新して、マネージドクラウドサービスとクラウド HA メカニズムを活用し始める必要があります。

以下のセクションでは、高可用性要件のワークロードを実行するためにオンプレミス環境に AWS Outposts ラックをデプロイするためのアーキテクチャパターン、アンチパターン、および推奨プラクティスを紹介します。これらのセクションではパターンとプラクティスを紹介しますが、設定や実装の詳細は説明しません。Outposts ラックのために環境を整え、AWS のサービスへの移行のためにアプリケーションを準備する際には、AWS Outposts ラックの FAQ とユーザーガイド、および Outposts ラックで実行されるサービスの FAQ とサービスドキュメントを読み、理解しておく必要があります。

トピック

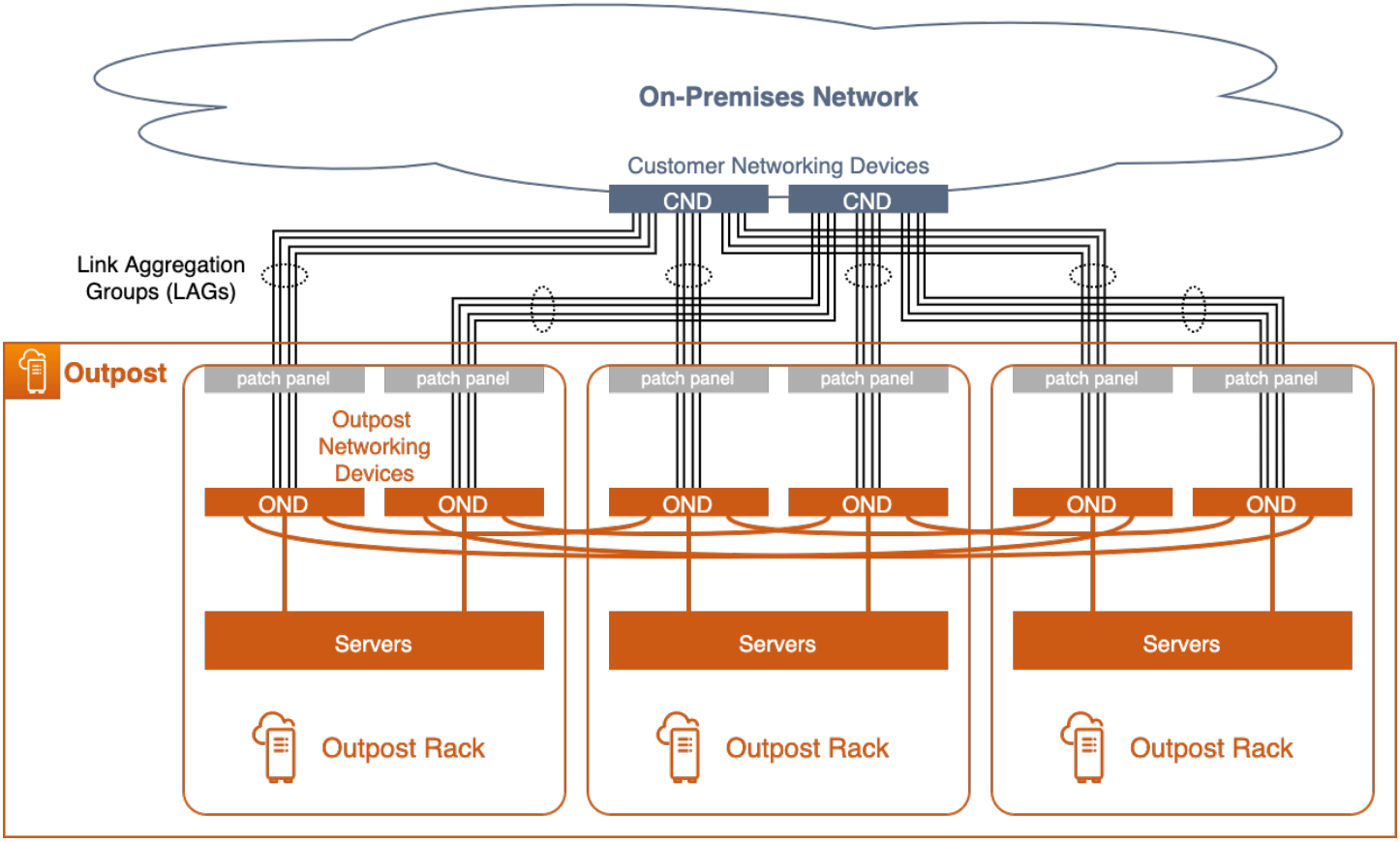
ネットワーク

Outpost のデプロイは、管理、モニタリング、サービス運用が適切に機能するために、アンカー AZ への耐障害性の高い接続が不可欠です。各 Outpost ラックに冗長ネットワーク接続を提供し、AWS クラウド内のアンカーポイントへの信頼性の高い接続を提供するようにオンプレミスネットワークをプロビジョニングする必要があります。また、Outpost で実行されているアプリケーションワークロードと、それらが通信する他のオンプレミスおよびクラウドシステムとの間のネットワークパスについても検討してください。このトラフィックをネットワーク内でどのようにルーティングしますか。

トピック

ネットワークアタッチメント

各 AWS Outposts ラックは、Outpost ネットワークデバイス (OND) と呼ばれる冗長なトップオブラックスイッチで構成されています。各ラックのコンピューティングサーバーとストレージサーバーは、両方の OND に接続します。各 OND をデータセンターのカスタマーネットワークデバイス (CND) と呼ばれる個別のスイッチに接続して、各 Outpost ラックにさまざまな物理パスと論理パスを提供する必要があります。OND は、光ファイバーケーブルと光トランシーバーを使用して 1 つ以上の物理接続で CND に接続します。物理接続は、リンク集約グループ (LAG) リンクで構成されます。



冗長ネットワークアタッチメントを備えたマルチラック Outpost

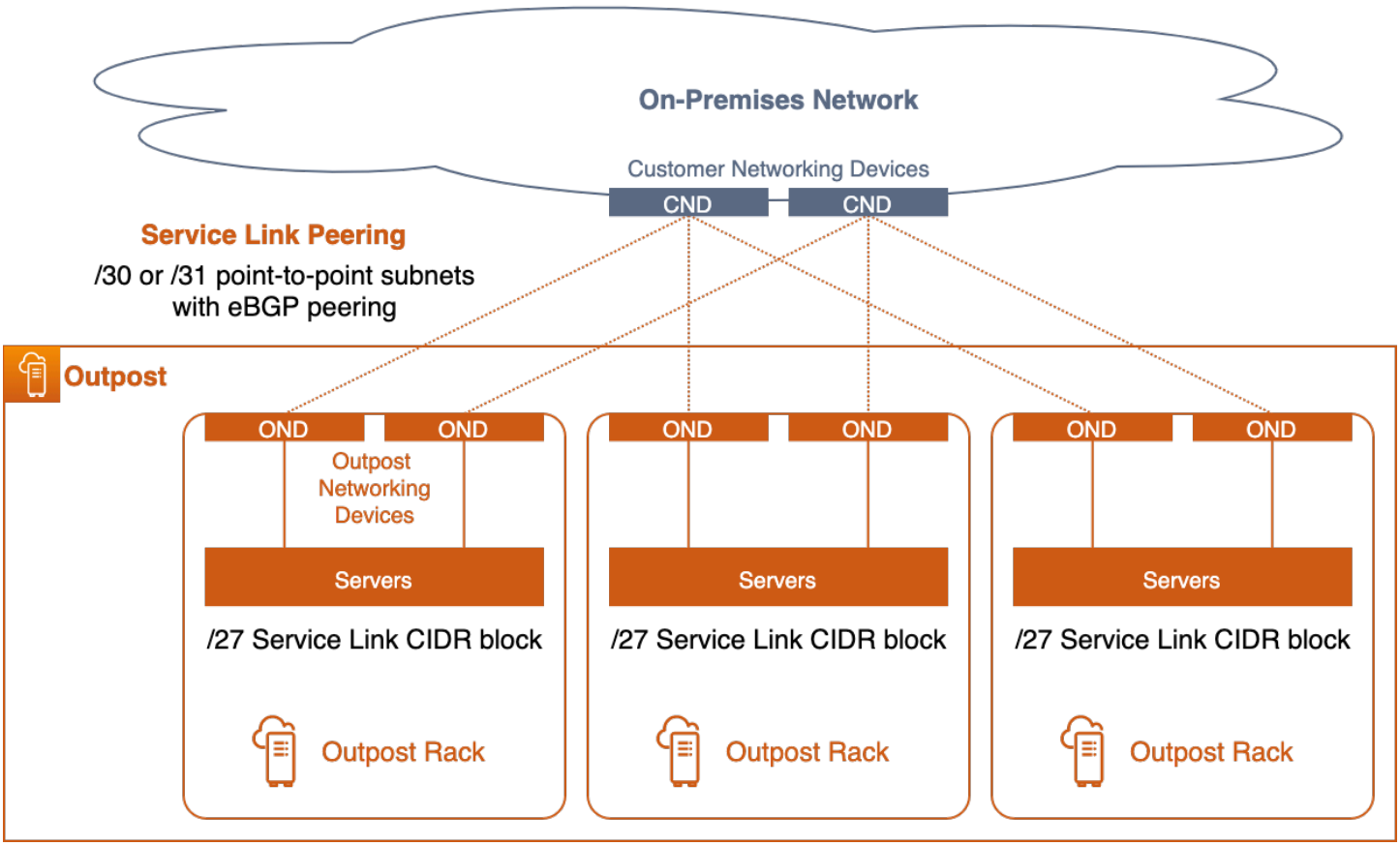
OND から CND へのリンクは、物理接続が 1 本の光ファイバーケーブルであっても、常に LAG で設定されます。リンクを LAG グループとして設定すると、論理グループに物理接続を追加してリン

ク帯域幅を増やすことができます。LAG リンクは IEEE 802.1q イーサネットトランクとして設定され、Outpost ネットワークとオンプレミスネットワーク間の分離されたネットワーキングを可能にします。

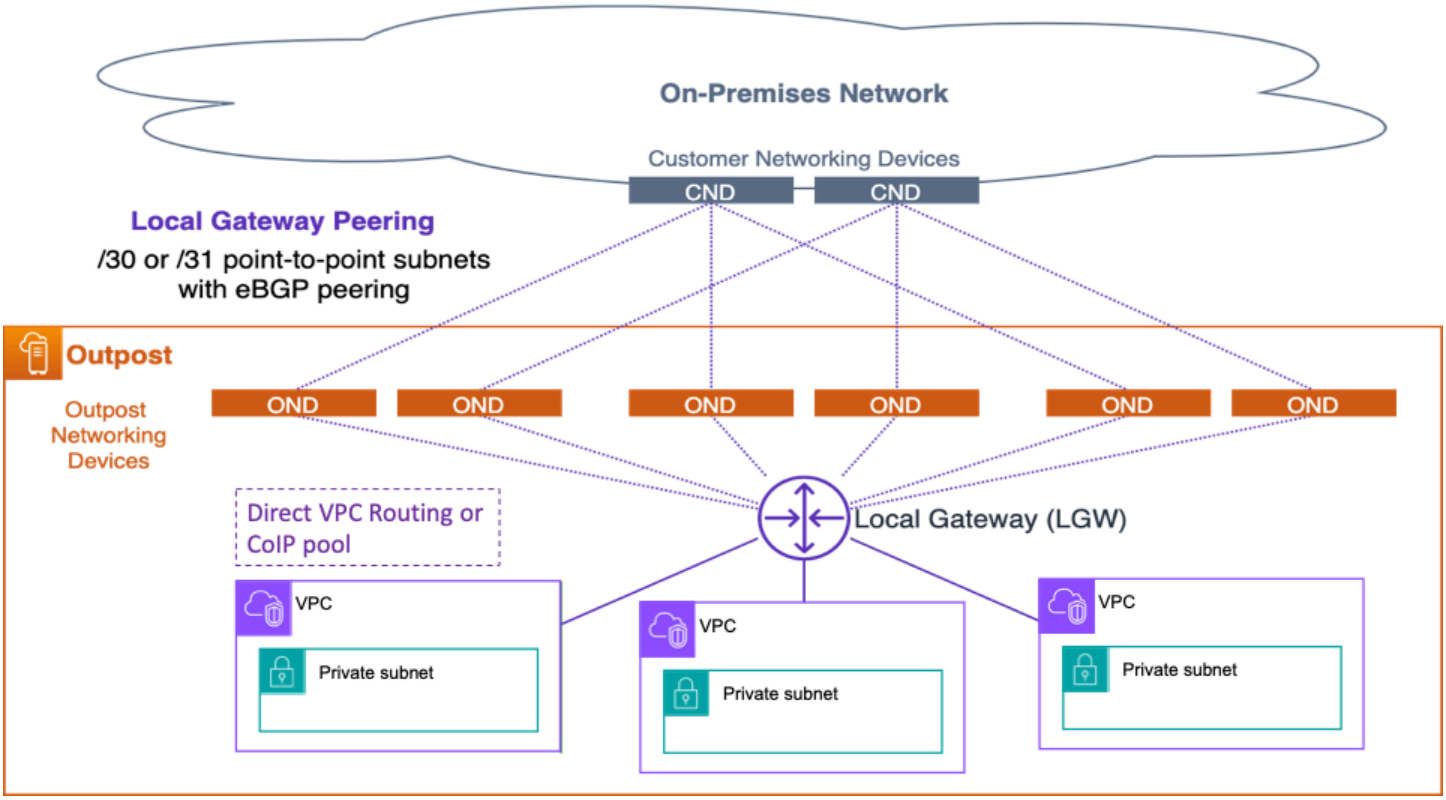
すべての Outpost には、カスタマーネットワークとの通信またはカスタマーネットワーク経由の通信が必要な、論理的に分離されたネットワークが少なくとも 2 つあります。

- サービスリンクネットワーク — サービスリンクの IP アドレスを Outpost サーバーに割り当て、オンプレミスネットワークとの通信を容易にし、サーバーがリージョン内の Outpost アンカーポイントに再接続できるようにします。単一の論理 Outposts に複数のラック実装がある場合は、各ラックにサービスリンク /26 CIDR を割り当てる必要があります。
- ローカルゲートウェイネットワーク — Outpost の VPC サブネットと Outpost ローカルゲートウェイ (LGW) 経由のオンプレミスネットワーク間の通信を可能にします。

これらの分離されたネットワークは、LAG リンクを介した一連のポイントツーポイント IP 接続によってオンプレミスネットワークに接続されます。OND から CND への各 LAG リンクは、分離されたネットワーク (サービスリンクと LGW) ごとに VLAN ID、ポイントツーポイント (/30 または /31) IP サブネット、および eBGP ピアリングで設定されます。ポイントツーポイントの VLAN とサブネットを含む LAG リンクは、レイヤ 2 セグメント化され、ルーティングされたレイヤ 3 接続と見なす必要があります。ルーティングされた IP 接続は、Outpost 上の分離されたネットワークとオンプレミスネットワーク間の通信を容易にする冗長な論理パスを提供します。



サービスリンクピアリング



ローカルゲートウェイピアリング

直接接続された CND スイッチでレイヤ 2 LAG リンク (およびその VLAN) を終了し、CND スイッチに IP インターフェイスと BGP ピアリングを設定する必要があります。データセンターのスイッチ間で LAG VLAN をブリッジしないでください。詳細については、AWS Outposts ユーザーガイドの「Network layer connectivity」(ネットワークレイヤーの接続性) を参照してください。

論理的なマルチラックの Outpost 内では、OND が冗長的に相互接続され、ラックとサーバー上で実行されているワークロード間の可用性の高いネットワーク接続を実現しています。AWS は、Outpost 内のネットワークの可用性を管理します。

ACE を使用しない可用性の高いネットワーク接続の推奨プラクティス

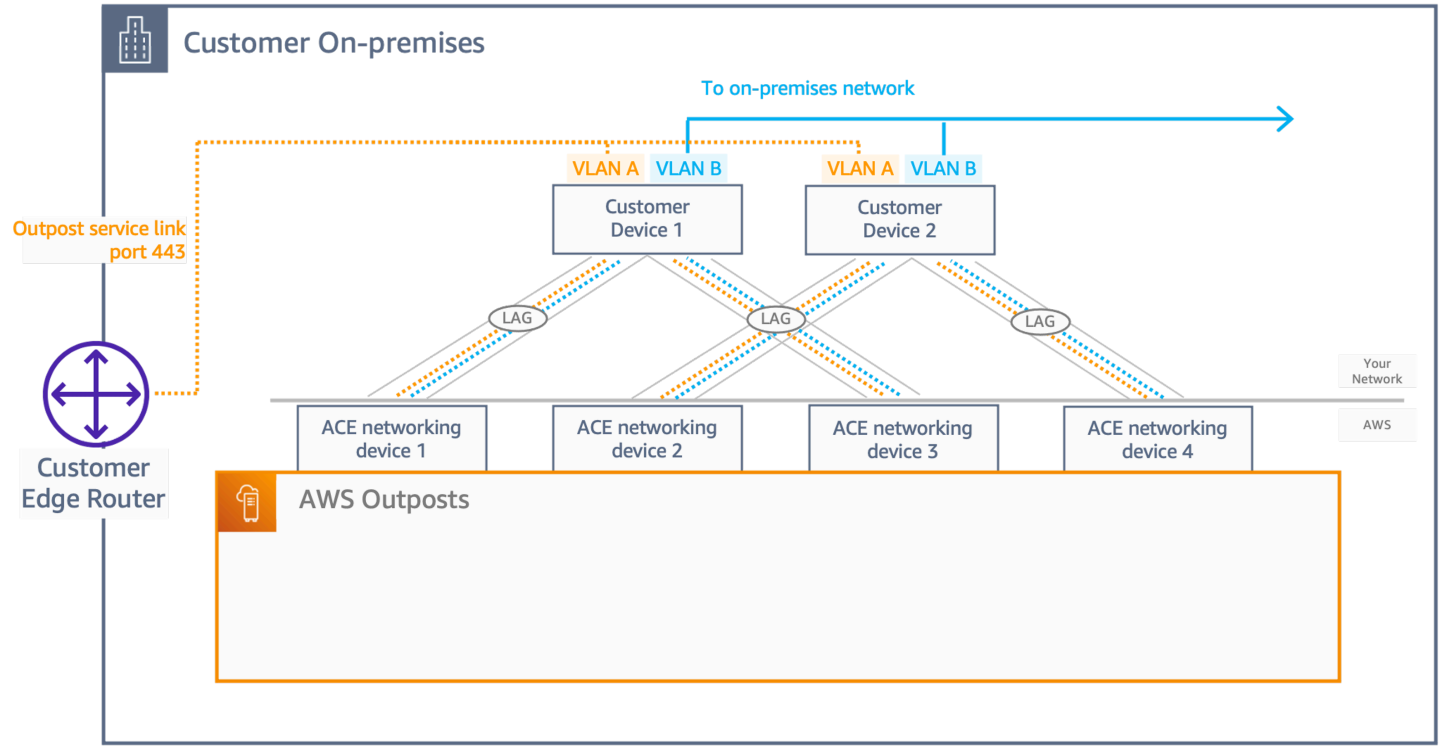
- Outpost ラック内の各 Outpost ネットワークデバイス (OND) を、データセンターの個別のカスタマーネットワークデバイス (CND) に接続します。
- 直接接続されたカスタマーネットワークデバイス (CND) スイッチ上でレイヤー 2 リンク、VLAN、レイヤー 3 IP サブネット、および BGP ピアリングを終了します。CND 間またはオンプレミスネットワーク経由で OND を CND VLAN にブリッジしないでください。
- リンク集約グループ (LAG) リンクを追加して、Outpost とデータセンター間で利用可能な帯域幅を増やします。両方の OND を経由するさまざまなパスの集約帯域幅に依存しないでください。

- 冗長 OND を経由するさまざまなパスを使用して、Outpost ネットワークとオンプレミスネットワーク間の耐障害性の高い接続を実現します。
- 最適な冗長性を実現し、中断することなく OND メンテナンスを行えるように、BGP アドバタイズとポリシーを次のように設定することをお勧めします。
  - お客様のネットワーク機器は、BGP 属性を変更せずに Outpost から BGP アドバタイズを受信し、BGP マルチパス/ロードバランシングを有効にして (お客様から Outpost への) 最適なインバウンドトラフィックフローを可能にする必要があります。メンテナンスが必要な場合に備えて、Outpost BGP プレフィックスに AS-Path への付加を使用して、トラフィックを特定の OND/アップリンクから移行します。お客様のネットワークは、AS-Path length 4 のルートよりも Outpost with AS-Path length 1 からのルートを優先する必要があります。つまり、AS-Path への付加に対応する必要があります。
  - お客様のネットワークは、Outpost 内のすべての OND に対して、同じ属性を持つ同じ BGP プレフィックスをアドバタイズする必要があります。デフォルトでは、Outpost ネットワークはすべてのアップリンク間で (お客様に向けた) アウトバウンドトラフィックの負荷分散を行います。Outpost 側では、メンテナンスが必要な場合にトラフィックを特定の OND から移行するために、ルーティングポリシーが使用されます。このトラフィック移行を実行し、中断することなくメンテナンスを実行するには、すべての OND でお客様側からの同じ BGP プレフィックスが必要です。お客様のネットワークでメンテナンスが必要な場合は、AS-Path への付加を使用して、特定のアップリンクまたはデバイスからのトラフィックを一時的に移行することをお勧めします。

ACE を使用した可用性の高いネットワーク接続の推奨プラクティス

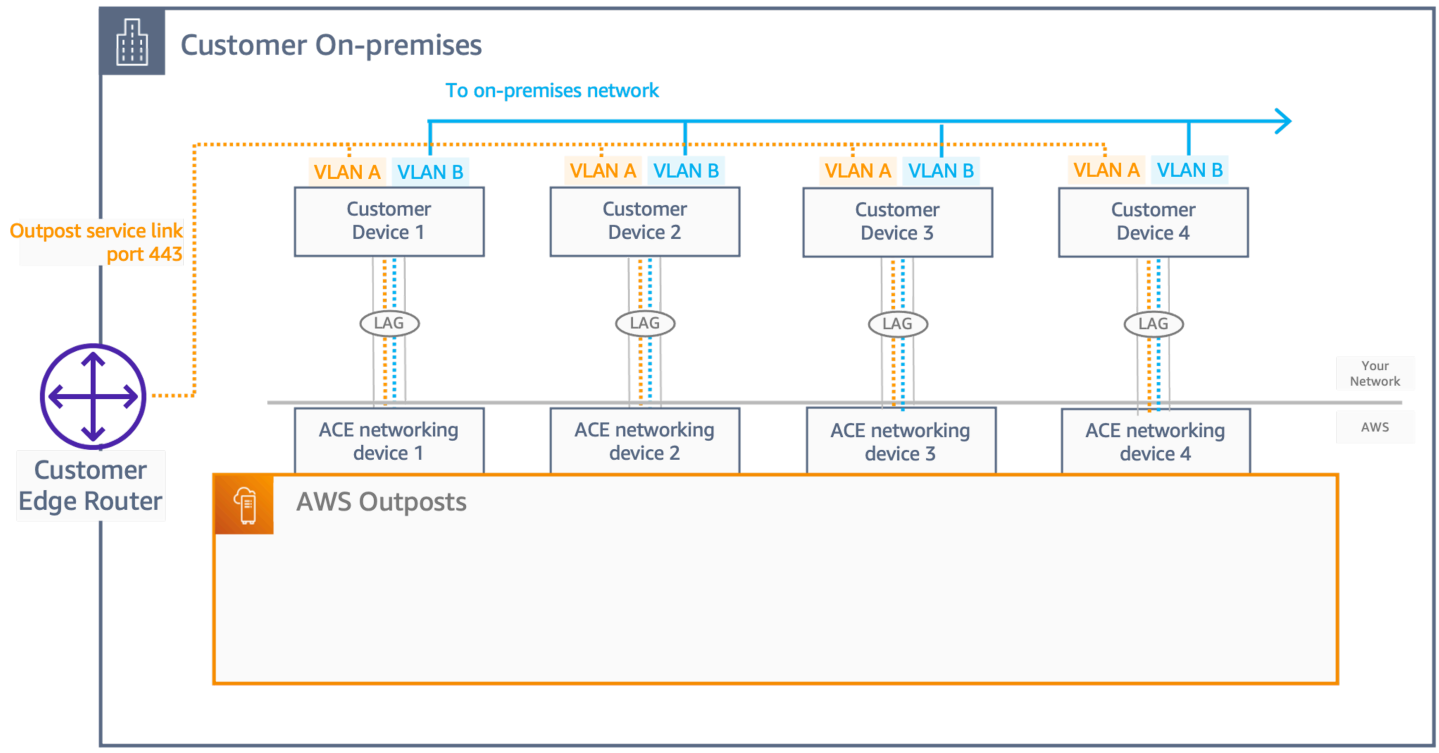
4 台以上のコンピューティングラックを備えたマルチラックデプロイでは、集約、コア、エッジ (ACE) ラックを使用する必要があります。これは、オンプレミスネットワークデバイスへのファイバーリンクの数を減らすためのネットワーク集約ポイントとして機能します。ACE ラックは各 Outposts ラックの OND への接続を提供するため、AWS は OND と ACE ネットワークデバイス間の VLAN インターフェイスの割り当てと設定を所有します。

サービスリンクとローカルゲートウェイネットワークの分離されたネットワークレイヤーは、ACE ラックを使用するかどうかにかかわらず引き続き必要です。これは、VLAN ポイントツーポイント (/30 または /31) IP サブネット、および分離された各ネットワークの eBGP ピアリング設定を持つためです。提案されたアーキテクチャは、次の 2 つのアーキテクチャのいずれかに従う必要があります。



2 つのカスタマーネットワークデバイス

- このアーキテクチャでは、お客様は 2 つのネットワークデバイス (CND) を使用して ACE ネットワークデバイスを相互接続し、冗長性を確保する必要があります。
- 各物理接続について、たとえそれが単一の物理ポートであっても、Outpost とデータセンター間の利用可能な帯域幅を増やすために LAG を有効にする必要があります。また、LAG は 2 つのネットワークセグメントを伝送し、2 つのポイントツーポイント VLAN (/30 または /31) と、ACE と CND 間の eBGP 設定を持ちます。
- 定常状態では、トラフィックは ACE レイヤーからカスタマーネットワークへの、またはカスタマーネットワークからの Equal-cost multipath (ECMP) パターンに従って負荷分散されます。ACE 全体でのトラフィック分散は 25% です。この動作を可能にするには、ACE と CND 間の eBGP ピアリングで BGP マルチパス/負荷分散が有効になっており、4 つの eBGP ピアリング接続で同じ BGP メトリクスを使用してカスタマープレフィックスがアナウンスされている必要があります。
- 最適な冗長性を実現し、中断のない OND メンテナンスを可能にするには、次の推奨事項に従うことをお勧めします。
  - カスタマーネットワークデバイスは、Outpost 内のすべての OND に対して、同じ属性を持つ同じ BGP プレフィックスをアドバタイズする必要があります。
  - カスタマーネットワークデバイスは、BGP 属性を変更せずに Outpost から BGP アドバタイズを受信し、BGP マルチパス/ロードバランシングを有効にする必要があります。



4 つのカスタマーネットワークデバイス

このアーキテクチャでは、ACE ネットワークデバイスを相互接続するための 4 つのネットワークデバイス (CND) を持つことになります。これにより、2 つの CND アーキテクチャに適用可能な VLAN、eBGP、ECMP などの冗長性と同一のネットワークロジックが提供されます。

アンカー接続

Outpost サービスリンクは、Outpost の親リージョンの特定のアベイラビリティーゾーン (AZ) にあるパブリックアンカーまたはプライベートアンカーのいずれか (両方ではない) に接続します。Outpost サーバーは、サービスリンク IP アドレスからアンカー AZ のアンカーポイントへのアウトバウンドサービスリンク VPN 接続を開始します。これらの接続は UDP と TCP ポート 443 を使用します。AWS は、リージョン内のアンカーポイントの可用性を確保する責任があります。

Outpost サービスリンク の IP アドレスがネットワーク経由でアンカー AZ のアンカーポイントに接続できることを確認する必要があります。サービスリンク IP アドレスは、オンプレミスネットワーク上の他のホストと通信する必要はありません。

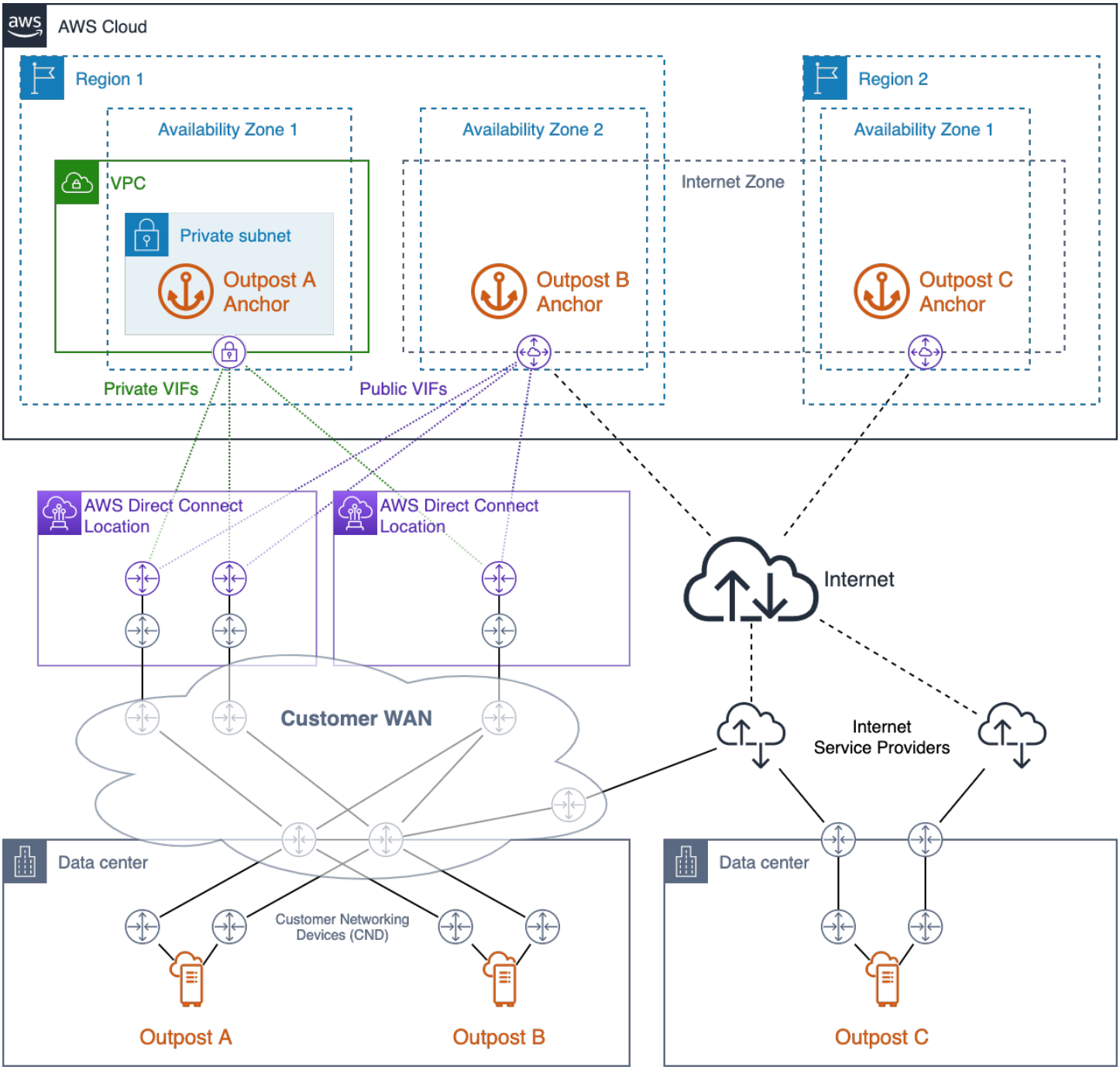
パブリックアンカーポイントは、リージョンのパブリック IP 範囲 (EC2 サービス CIDR ブロック内) にあり、インターネットまたは AWS Direct Connect (DX) パブリック仮想インターフェイス (VIF) 経由でアクセスできます。パブリックアンカーポイントを使用すると、サービスリンクトラフィックは

パブリックインターネット上のアンカーポイントに正常に到達できる任意のパスにルーティングできるため、より柔軟にパスを選択できます。

プライベートアンカーポイントを使用すると、IP アドレス範囲をアンカー接続に使用できます。プライベートアンカーポイントは、お客様が割り当てた IP アドレスを使用して、専用 VPC 内のプライベートサブネットに作成されます。VPC は Outpost リソースを所有する AWS アカウントで作成され、VPC が使用可能で、適切に設定されていることを確認する必要があります。AWSOrigamiServiceGateway Organizations でセキュリティコントロールポリシー (SCP) を使用して、ユーザーがその仮想プライベートクラウド (VPC) を削除できないようにします。プライベートアンカーポイントには、Direct Connect プライベート VIF を使用してアクセスする必要があります。

Outpost とリージョン内のアンカーポイントの間には冗長なネットワークパスを用意し、接続は複数の場所にある別々のデバイスで終了するようにしてください。動的ルーティングは、接続またはネットワークデバイスに障害が発生した場合に、トラフィックを自動的に代替パスに再ルートするように設定する必要があります。1 つの WAN パスに障害が発生しても残りの WAN パスに負荷がかからないように、十分なネットワーク容量をプロビジョニングする必要があります。

次の図は、AWS Direct Connect とパブリックインターネット接続を使用する、アンカー AZ への冗長ネットワークパスがある 3 つの Outposts を示しています。Outpost A と Outpost B は、同じリージョンの異なるアベイラビリティーゾーンにアンカーされています。Outpost A は、リージョン 1 の AZ 1 のプライベートアンカーポイントに接続しています。Outpost B はリージョン 1 の AZ 2 のパブリックアンカーポイントに接続しています。Outpost C はリージョン 2 の AZ 1 のパブリックアンカーに接続します。



AWS Direct Connect とパブリックインターネットアクセスとの高可用性アンカー接続

Outpost A には、プライベートアンカーポイントに到達するための冗長ネットワークパスが 3 つあります。1 つの Direct Connect の場所では、冗長な Direct Connect 回路を通じて 2 つのパスを使用できます。3 番目のパスは、2 つ目の Direct Connect の場所にある Direct Connect 回路を経由して利用できます。この設計では、Outpost A のサービスリンクトラフィックをプライベートネットワー

ク上に維持し、パスの冗長性を確保することで、いずれかの Direct Connect 回線の障害や Direct Connect の場所全体の障害にも対応できます。

Outpost B には、パブリックアンカーポイントに到達するための冗長ネットワークパスが 4 つあります。3 つのパスは、Direct Connect 回線と Outpost A が使用する場所にプロビジョニングされたパブリック VIF を介して利用できます。4 つ目のパスは、お客様の WAN とパブリックインターネットを介して利用できます。Outpost B のサービスリンクトラフィックは、パブリックインターネット上のアンカーポイントに正常に到達できる任意のパスを介してルートできます。Direct Connect パスを使用すると、レイテンシーがより安定し、帯域幅の可用性が高くなる可能性があります。一方、パブリックインターネットパスはディザスタリカバリ (DR) や帯域幅増強のシナリオに使用できます。

Outpost C には、パブリックアンカーポイントに到達するための冗長ネットワークパスが 2 つあります。Outpost C は、Outpost A および Outpost B とは異なるデータセンターでデプロイされています。Outpost C のデータセンターには、お客様の WAN に接続する専用回線はありません。代わりに、データセンターには 2 つの異なるインターネットサービスプロバイダー (ISP) が提供する冗長インターネット接続があります。Outpost C のサービスリンクトラフィックは、いずれかの ISP ネットワークを経由してルートされ、パブリックインターネット上のアンカーポイントに到達できます。この設計により、利用可能なあらゆるパブリックインターネット接続でサービスリンクトラフィックを柔軟にルートできます。ただし、エンドツーエンドのパスは、帯域幅の可用性とネットワーク遅延が変動するパブリックサードパーティーネットワークに依存しています。

Outpost とそのサービスリンクアンカーポイント間のネットワークパスは、次の帯域幅の仕様を満たしている必要があります。

- 500 Mbps - Outpost ラックあたり 1 Gbps の使用可能帯域幅 (例えば、3 ラックの場合、1.5 ~ 3 Gbps の使用可能帯域幅)

高可用性アンカー接続の推奨プラクティス

- 各 Outpost とリージョン内のアンカーポイントとの間に冗長なネットワークパスをプロビジョニングします。
- Direct Connect (DX) パスを使用して、レイテンシーと帯域幅の可用性を制御します。
- Outpost サービスリンク CIDR ブロックから親リージョンの EC2 IP アドレス範囲への TCP および UDP ポート 443 が開いている (アウトバウンド) ことを確認します。すべてのネットワークパスでポートが開いていることを確認します。
- リージョンの CIDR 範囲のサブセットを使用している場合は、ファイアウォールの Amazon EC2 IP アドレス範囲を追跡します。

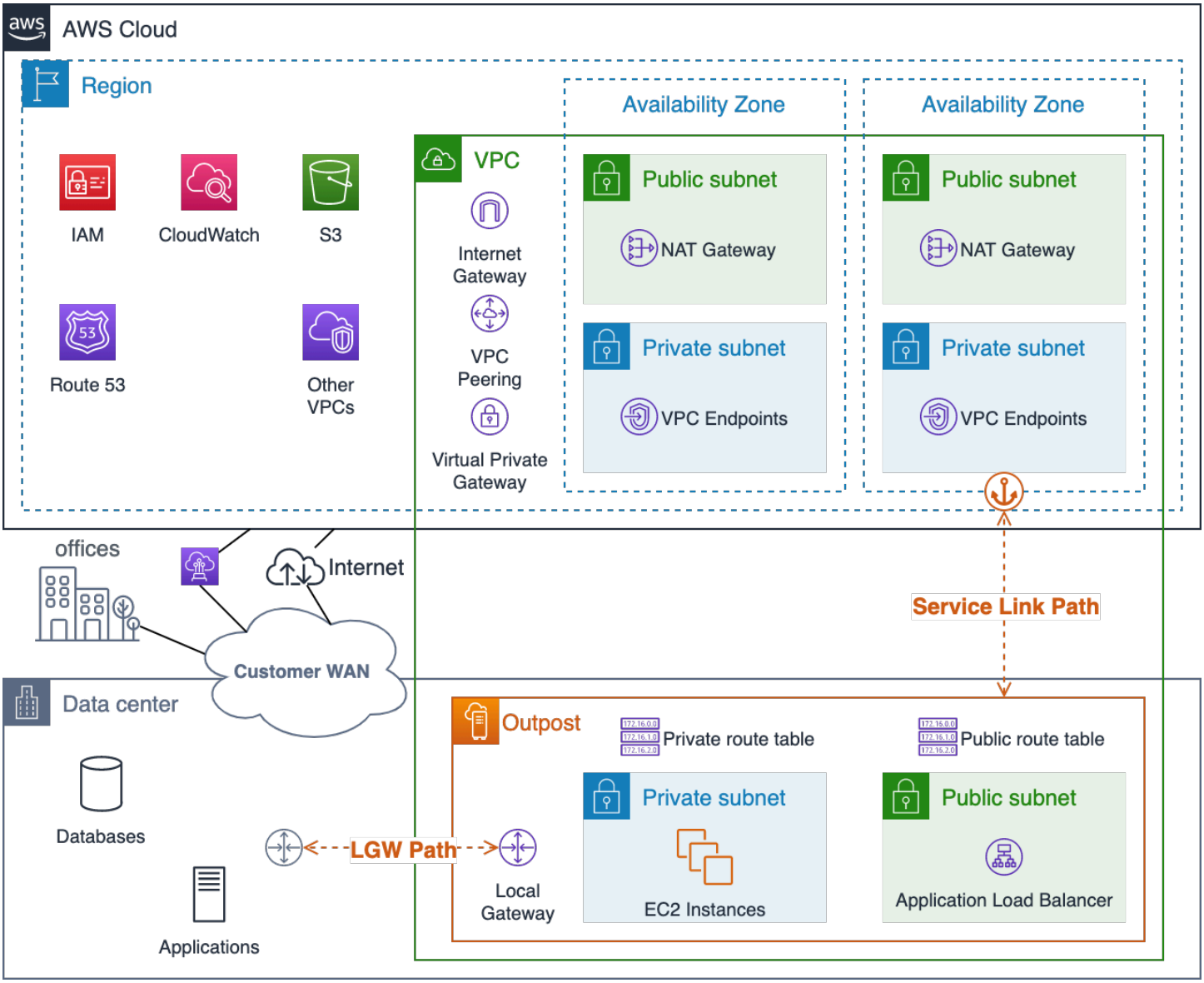
- 各パスが帯域幅の可用性とレイテンシーの要件を満たしていることを確認します。
- 動的ルーティングを使用して、ネットワーク障害を回避するトラフィックのリダイレクトを自動化します。
- 計画した各ネットワークパスでサービスリンクトラフィックのルーティングをテストして、パスが想定どおりに機能することを確認します。

アプリケーション/ワークロードのルーティング

Outpost からアプリケーションワークロードには、次の 2 つのパスがあります。

- サービスリンクパス: MTU を 1,300 バイトに制限することに加え、アプリケーショントラフィックが Outposts コントロールプレーントラフィックと競合することを考慮してください。
- ローカルゲートウェイ (LGW) パス: お客様のローカルネットワークが、オンプレミスと AWS リージョンの両方のアプリケーションへのアクセスを許可することを検討します。

Outpost サブネットルートテーブルを設定して、宛先ネットワークに到達するまでのパスを制御します。LGW を指すルートは、トラフィックをローカルゲートウェイからオンプレミスネットワークに転送します。インターネットゲートウェイ、NAT ゲートウェイ、仮想プライベートゲートウェイ、TGW など、リージョン内のサービスとリソースを指すルートは、サービスリンクを使用してこれらのターゲットに到達します。同じ Outpost にある複数の VPC ピアリング接続を使用している場合、VPC 間のトラフィックは Outpost に残り、リージョンに戻るサービスリンクは使用しません。VPC ピアリングの詳細については、「Amazon VPC ユーザーガイド」の「VPC ピアリングを使用して VPC を接続する」を参照してください。



Outpost サービスリンクと LGW ネットワークパスの視覚化

アプリケーションルーティングを計画する際は、ネットワーク障害時の通常の運用と制限されたルーティングとサービスの可用性の両方を考慮するように注意する必要があります。Outpost がリージョンから切断されている場合、サービスリンクパスは利用できません。

Outpost LGW と重要なオンプレミスのアプリケーション、システム、およびユーザーとの間には、さまざまなパスをプロビジョニングし、動的ルーティングを設定する必要があります。ネットワークパスが冗長化されていると、障害が発生してもネットワークがトラフィックをルートできるようになり、部分的なネットワーク障害が発生しても、オンプレミスのリソースが Outpost で実行されているワークロードと通信できるようになります。

Outpost VPC ルート設定は静的です。サブネットルーティングテーブルは、AWS Management Console、CLI、API、およびその他の Infrastructure as Code (IaC) ツールを使用して設定しますが、切断イベント中はサブネットルーティングテーブルを変更できません。ルートテーブルを更新するには、Outpost とリージョン間の接続を再確立する必要があります。通常の運用には、切断イベント時に使用する予定と同じルートを使用してください。

Outpost 上のリソースは、サービスリンクとリージョン内のインターネットゲートウェイ (IGW)、またはローカルゲートウェイ (LGW) パスを介してインターネットにアクセスできます。LGW パスとオンプレミスネットワークを介してインターネットトラフィックをルーティングすると、既存のオンプレミスのインターネット入出力ポイントを使用でき、リージョン内の IGW へのサービスリンクパスを使用する場合と比較して、レイテンシーが低く、MTU が高くなり、AWS のデータ送信料金が削減されます。

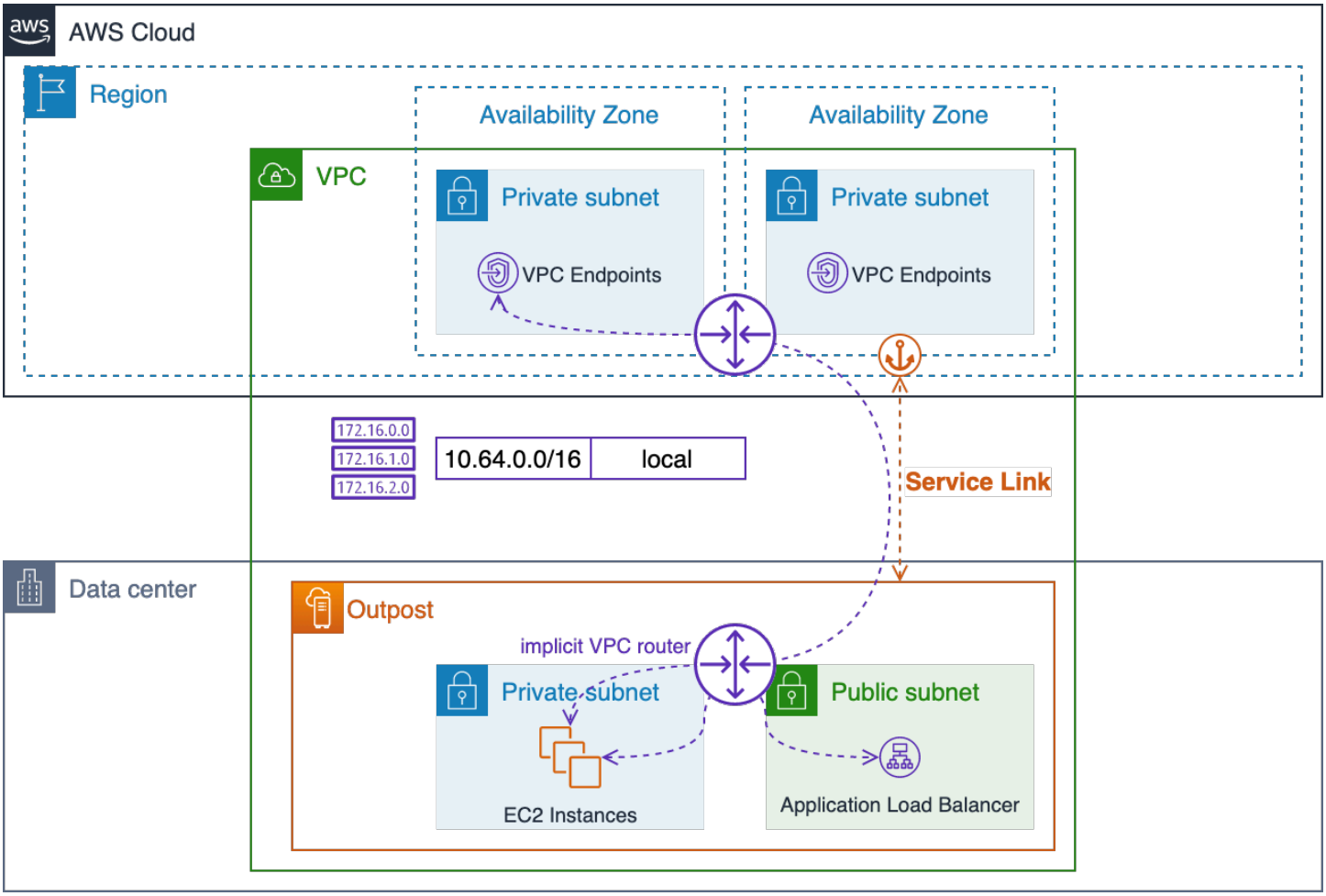
アプリケーションをオンプレミスで実行する必要があり、パブリックインターネットからアクセスできるようにする必要がある場合は、アプリケーショントラフィックをオンプレミスのインターネット接続経由で LGW にルーティングして Outpost 上のリソースに到達する必要があります。

Outpost のサブネットは、リージョンのパブリックサブネットのように設定できますが、ほとんどのユースケースでは望ましくない方法です。インバウンドのインターネットトラフィックは AWS リージョン を経由して入ってきて、サービスリンクを経由して Outpost で実行されているリソースにルーティングされます。

応答トラフィックは、次にサービスリンクを介してルーティングされ、AWS リージョン のインターネット接続を経由して戻ります。このトラフィックパターンではレイテンシーが発生する可能性があり、トラフィックが Outpost に向かう途中でリージョンを離れ、リターントラフィックがリージョンを経由してインターネットに出るときに、データ出力の料金が発生します。アプリケーションをリージョンで実行できる場合は、そのリージョンで実行するのが最適です。

(同じ VPC 内の) VPC リソース間のトラフィックは、常にローカルの VPC CIDR ルートをたどり、暗黙的 VPC ルーターによってサブネット間でルーティングされます。

例えば、Outpost で実行されている EC2 インスタンスとリージョン内の VPC エンドポイントの間のトラフィックは、常にサービスリンクを介してルーティングされます。



暗黙的ルーターを介したローカル VPC ルーティング

アプリケーション/ワークロードのルーティングの推奨プラクティス

- 可能な場合は、サービスリンクパスの代わりにローカルゲートウェイ (LGW) パスを使用します。
- LGW パスを介してインターネットトラフィックをルーティングします。
- Outpost サブネットルーティングテーブルを標準のルートセットで設定します。これらは通常の運用と切断イベントの両方に使用されます。
- Outpost LGW と重要なオンプレミスアプリケーションリソース間の冗長ネットワークパスをプロビジョニングします。動的ルーティングを使用して、オンプレミスのネットワーク障害を回避するトラフィックのリダイレクトを自動化します。

コンピューティング

AWS リージョンにおける Amazon EC2 の容量は無限のように見えますが、Outposts の容量には限りがあります。Outposts デプロイのコンピューティングキャパシティの計画と管理はお客様の責任となります。

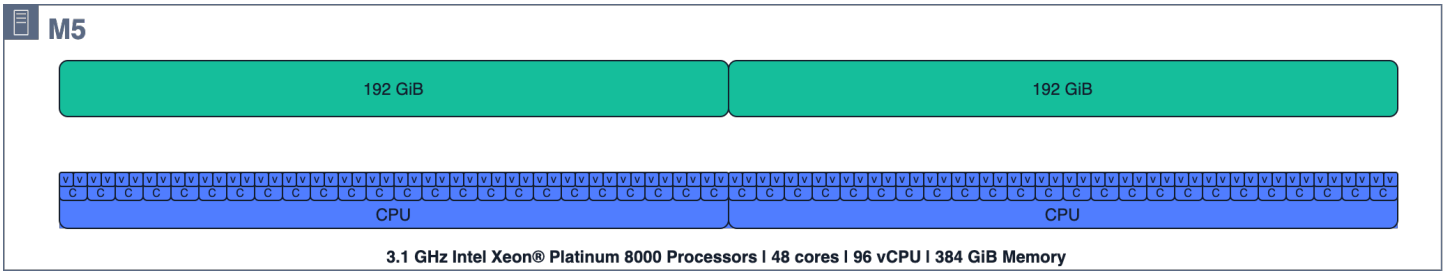
トピック

キャパシティプランニング

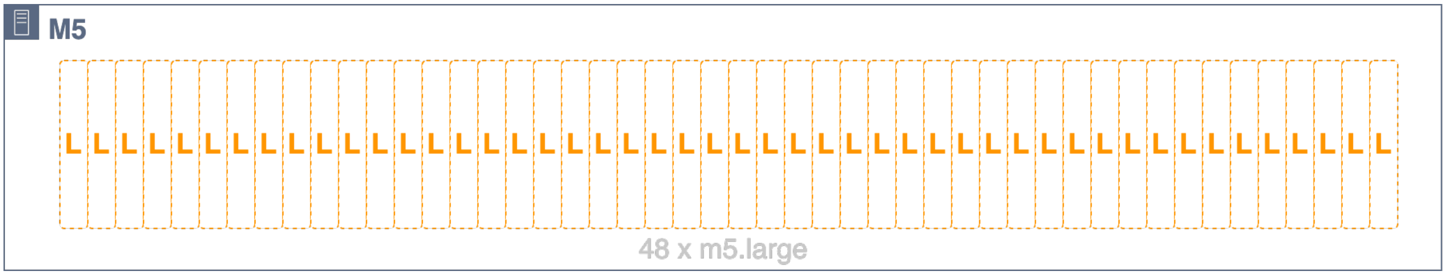
AWS リージョンにおける Amazon EC2 の容量は無限のように見えますが、Outposts の容量には限りがあり、注文したコンピューティング容量の合計によって制約されます。Outposts デプロイのコンピューティングキャパシティの計画と管理はお客様の責任となります。N+M 可用性モデルをサポートするのに十分なコンピューティングキャパシティを注文する必要があります。N は必要なサーバー数、M はサーバー障害に対応するためにプロビジョニングされたスペアサーバーの数です。N+1 と N+2 が最も一般的な可用性レベルです。

各ホスト (C5、M5、R5 など) は、単一ファミリーの EC2 インスタンスをサポートします。EC2 コンピューティングサーバーでインスタンスを起動する前に、各サーバーが提供する EC2 インスタンスサイズを指定するスロッティングレイアウトを準備する必要があります。AWS は、要求されたスロッティングレイアウトで各サーバーを構成します。

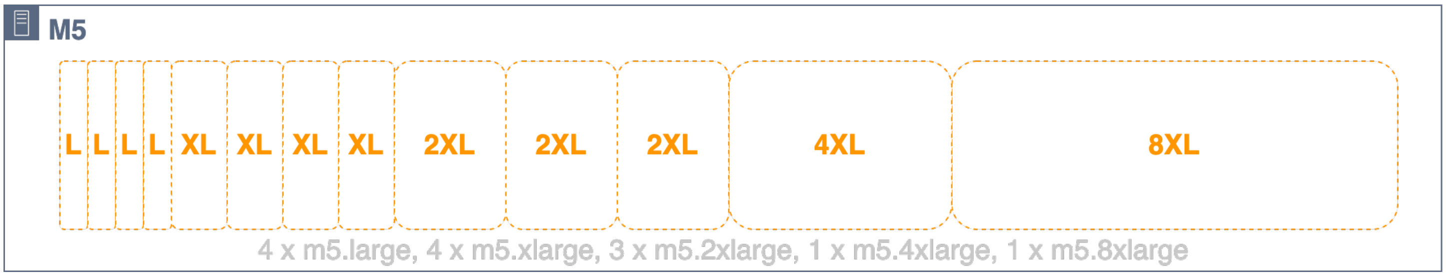
ホストは、すべてのスロットが同じインスタンスサイズ (48 個の `m5.large` スロットなど) である同種スロットでも、インスタンスタイプが混在する異種スロット (4 つの `m5.large`、4 つの `m5.xlarge`、3 つの `m5.2xlarge`、1 つの `m5.4xlarge`、および 1 つの `m5.8xlarge` など) でもかまいません。これらのスロッティング設定の視覚化については、次の 3 つの図を参照してください。



m5.24xlarge ホストのコンピューティングリソース



48 個の *m5.large* スロットに同種スロットされた *m5.24xlarge* ホスト



4 つの *m5.large*、4 つの *m5.xlarge*、3 つの *m5.2xlarge*、1 つの *m5.4xlarge*、および 1 つの *m5.8xlarge* スロットに異種スロットされた *m5.24xlarge* ホスト

ホストの全容量をスロットする必要はありません。空き容量が割り当てられていないホストにスロットを追加できます。AWS Outposts のキャパシティ管理 API または UI を使用して新しいキャパシティタスクを作成することで、スロッティングレイアウトを変更できます。詳細については、「ラックの AWS Outposts ユーザーガイド」の「[Capacity management for AWS Outposts]」を参照してください。特定のスロットが実行中のインスタンスによって占有されているため、新しいスロットレイアウトを適用できない場合は、新しいキャパシティタスクを完了するために特定のインスタンスをシャットダウンまたは再起動する必要がある場合があります。`CreateCapacityTask` API を使用すると、指定された Outpost ID に存在する各インスタンスサイズの数を表すことができ、実行中のインスタンスのためにタスクを完了できない場合は、リクエストを満たすために停止する必要があるインスタンスを返します。この時点で、返されたインスタンスの 1 つを停止したくない場合は、「N」個の追加オプションを表示することをオプションで指定できます。また、キャパシティタスクタスクのリクエストを満たすためにシャットダウンするインスタンスとして提案しない EC2 インスタンス ID、EC2 インスタンスタグ、アカウント、またはサービスを指定することもできます。使用するオプションを選択したら、Dry Run パラメータを使用して提案された変更を検証し、実装する前に潜在的な影響を理解することをお勧めします。

すべてのホストは、プロビジョニングされたスロットを Outpost の EC2 容量プールに提供し、特定のインスタンスタイプとサイズのすべてのスロットは 1 つの EC2 容量プールとして管理されます。

例えば、m5.large、m5.xlarge、m5.2xlarge、m5.4xlarge、および m5.8xlarge スロットを持つ以前の異種スロットホストは、これらのスロットを 5 つの EC2 容量プール、つまりインスタンスタイプとサイズごとに 1 つのプールに分配します。これらのプールは複数のホストに分散される可能性があるため、ワークロードの高可用性を実現するためにはインスタンス配置を検討する必要があります。

N+M ホストの可用性を考慮して予備の容量を計画する際には、ホストスロッティングと EC2 容量プールを考慮することが重要です。AWS は、ホストの失敗や機能低下を検出し、失敗したホストを交換するためにサイト訪問をスケジュールします。EC2 容量プールは、Outpost 内の各インスタンスファミリー (N+1) の少なくとも 1 台のサーバーの障害に耐えるように設計する必要があります。この最低限のホスト可用性により、ホストが失敗した場合やサービスを停止する必要が生じた場合に、同じファミリーの残りのホストのスペアスロットで、失敗したインスタンスや劣化したインスタンスを再起動できます。

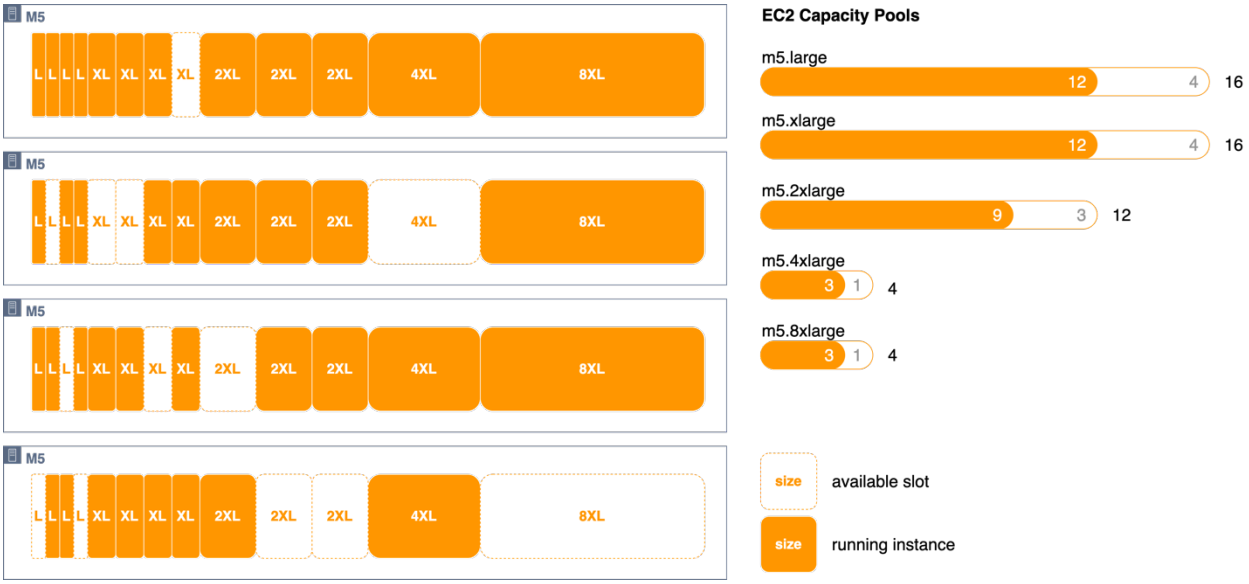
同じスロットのホストや、同じスロッティングレイアウトを持つ異種スロットのホストのグループがある場合、N+M の可用性の計画はシンプルです。すべてのワークロードを実行するのに必要なホストの数 (N) を計算し、障害やメンテナンス中のサーバーの可用性に関する要件を満たすために (M) 台のホストを追加するだけです。

NUMA 境界のため、次のスロッティング設定は使用できません。

- 3 m5.8xlarge
- 1 つの m5.16xlarge および 1 つの m5.8xlarge

AWS アカウント チームに連絡して、計画された AWS Outposts ラックスロッティング設定を検証してください。

次の図では、4 台の m5.24xlarge ホストが同じスロッティングレイアウトで異種スロットされています。4 台のホストは 5 個の EC2 容量プールを作成します。これら 4 台のホストで実行されているインスタンスの N+1 の可用性を維持するために、各プールは最大の使用率 (75%) で稼働しています。いずれかのホストが失敗しても、残りのホストで失敗したインスタンスを再起動するための十分な余裕があります。



EC2 ホストスロット、実行中のインスタンス、スロットプールの視覚化

ホストのスロット構成が同じでない、より複雑なスロッティングレイアウトの場合は、EC2 容量プールごとに N+M の可用性を計算する必要があります。次の式を使用して、(特定の EC2 容量プールにスロットを割り当てる) ホストが失敗しても、残りのホストで実行中のインスタンスを処理できるホストの台数を計算できます。

$$M = \left\lceil \frac{poolSlots_{available}}{serverSlots_{max}} \right\rceil$$

コードの説明は以下のとおりです。

- $poolSlots_{available}$ は、特定の EC2 容量プールで使用可能なスロットの数です (プール内のスロットの総数から実行中のインスタンスの数を引いたもの)
- $serverSlots_{max}$ は、任意のホストが特定の EC2 容量プールに提供するスロットの最大数
- M は、失敗しても残りのホストで実行中のインスタンスを実行できるホストの台数

例: Outpost には、1 つの `m5.2xlarge` 容量プールにスロットを提供するホストが 3 台あります。最初のホストは 4 スロット、2 番目のホストは 3 スロット、3 番目のホストは 2 スロットを提供します。Outpost の `m5.2xlarge` インスタンスプールの合計容量は 9 スロット (4 + 3 + 2) です。Outpost には 4 つの実行中の `m5.2xlarge` インスタンスがあります。失敗しても、残りのホストで実行中のインスタンスを実行できるホストは何台ですか。

$$poolSlots_{available} = total\ capacity - running\ instances = 9 - 4 = 5$$

$$serverSlots_{max} = \max([4, 3, 2]) = 4$$

$$M = \left[\frac{poolSlots_{available}}{serverSlots_{max}}\right] = \left[\frac{5}{4}\right] = [1.25] = 1$$

回答: いずれのホストを失っても、残りのホストで実行中のインスタンスを引き続き使用できます。

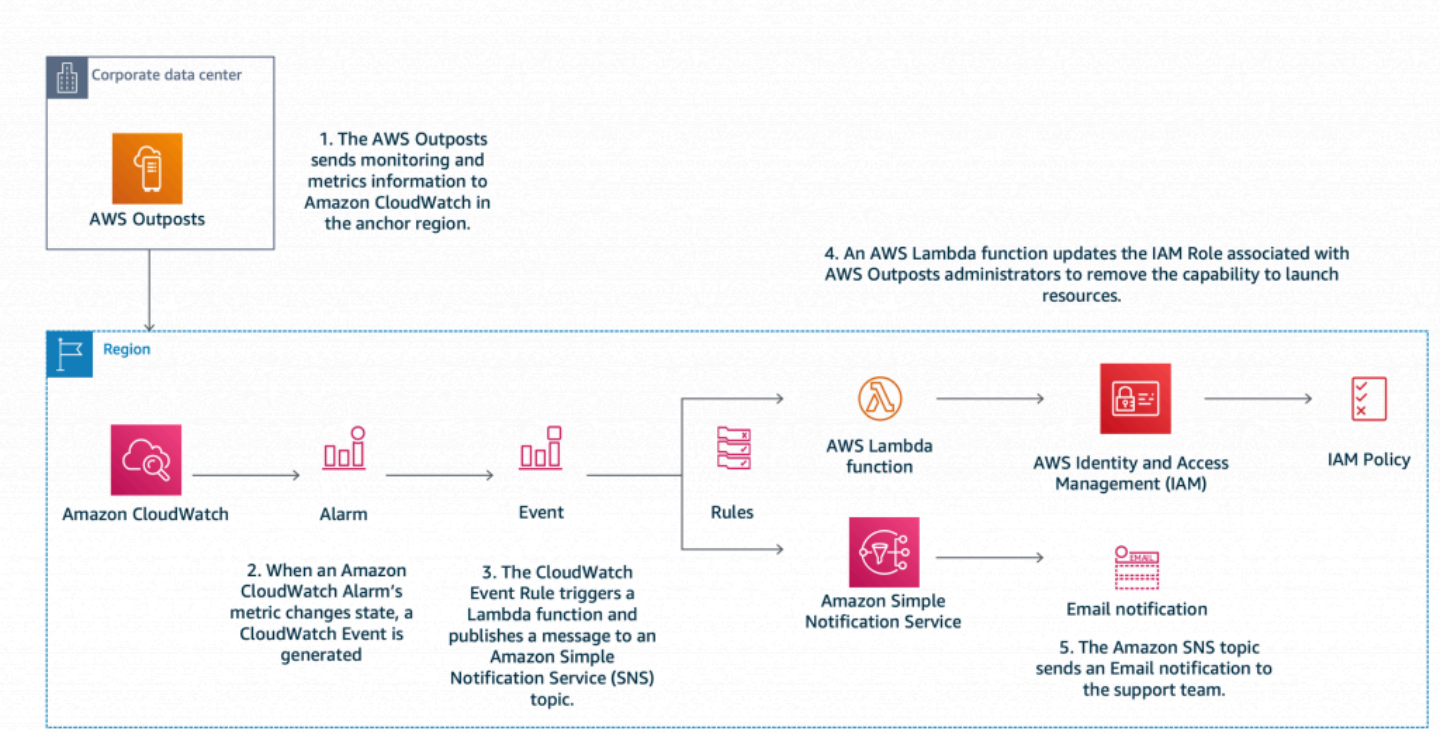
コンピューティングキャパシティプランニングの推奨プラクティス

- Outpost の各 EC2 容量プールに N+M の冗長性を持たせるように、コンピューティングキャパシティのサイズを設定します。
  - 同種または同一の異種スロットのサーバーに N+M 台のサーバーをデプロイします。
  - 各 EC2 容量プールの N+M 可用性を計算し、各プールが可用性要件を満たしていることを確認します。

キャパシティ管理

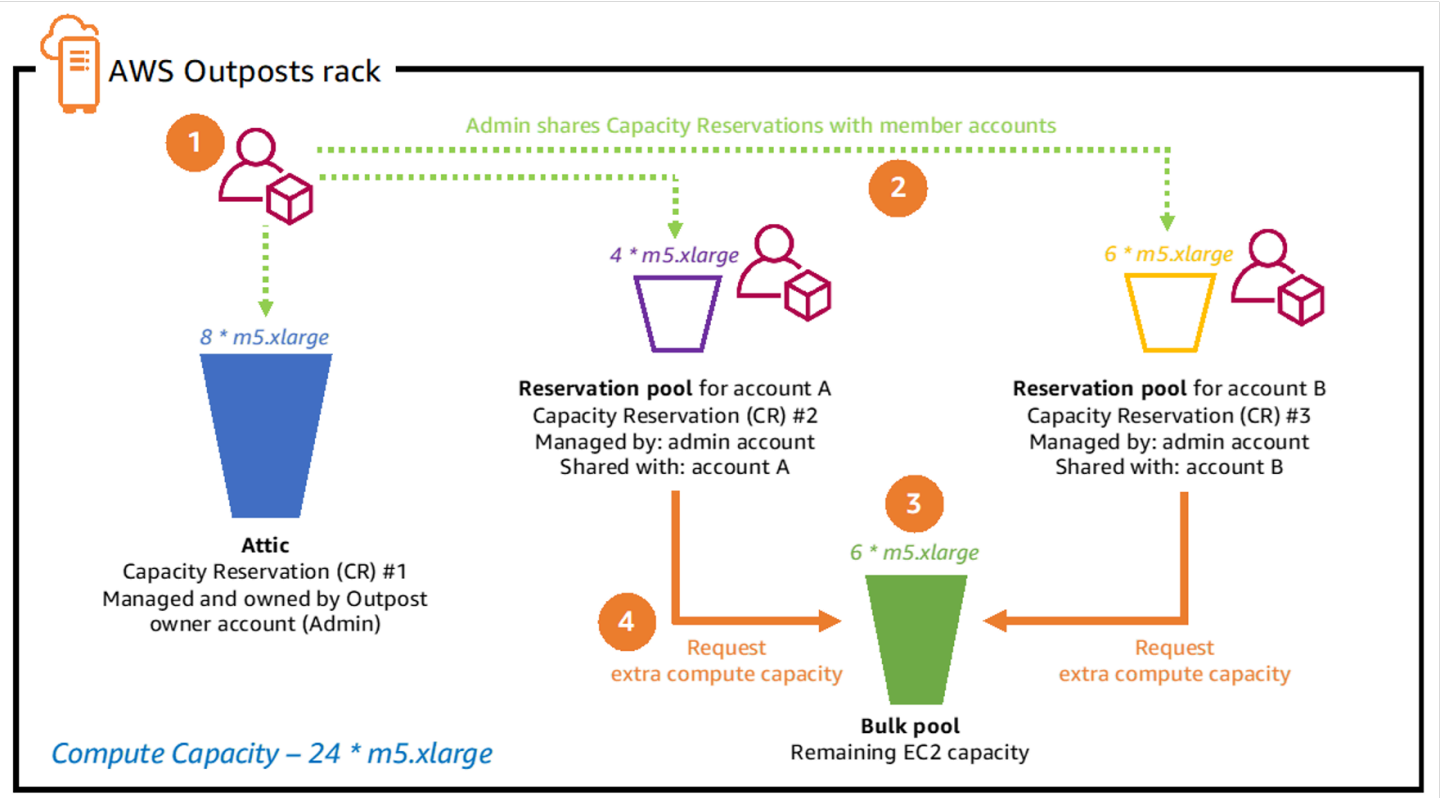
Outpost EC2 インスタンスプールの使用率を AWS Management Console で、および Amazon CloudWatch メトリクスを介してモニタリングできます。Outposts のスロッティングレイアウトを取得または変更するには、エンタープライズサポートに連絡してください。

同じインスタンスの自動リカバリと EC2 自動スケーリングのメカニズムを使用して、サーバー障害やメンテナンスイベントの影響を受けたインスタンスを復旧または交換します。Outpostの容量をモニタリングおよび管理して、サーバーの障害に対応できる十分な予備の容量を常に確保する必要があります。「Managing your AWS Outposts capacity using Amazon CloudWatch and AWS Lambda」ブログ記事では、AWS CloudWatch と AWS Lambda を組み合わせて Outpost のキャパシティを管理し、インスタンスの可用性を維持する方法を示す実践的なチュートリアルを提供しています。

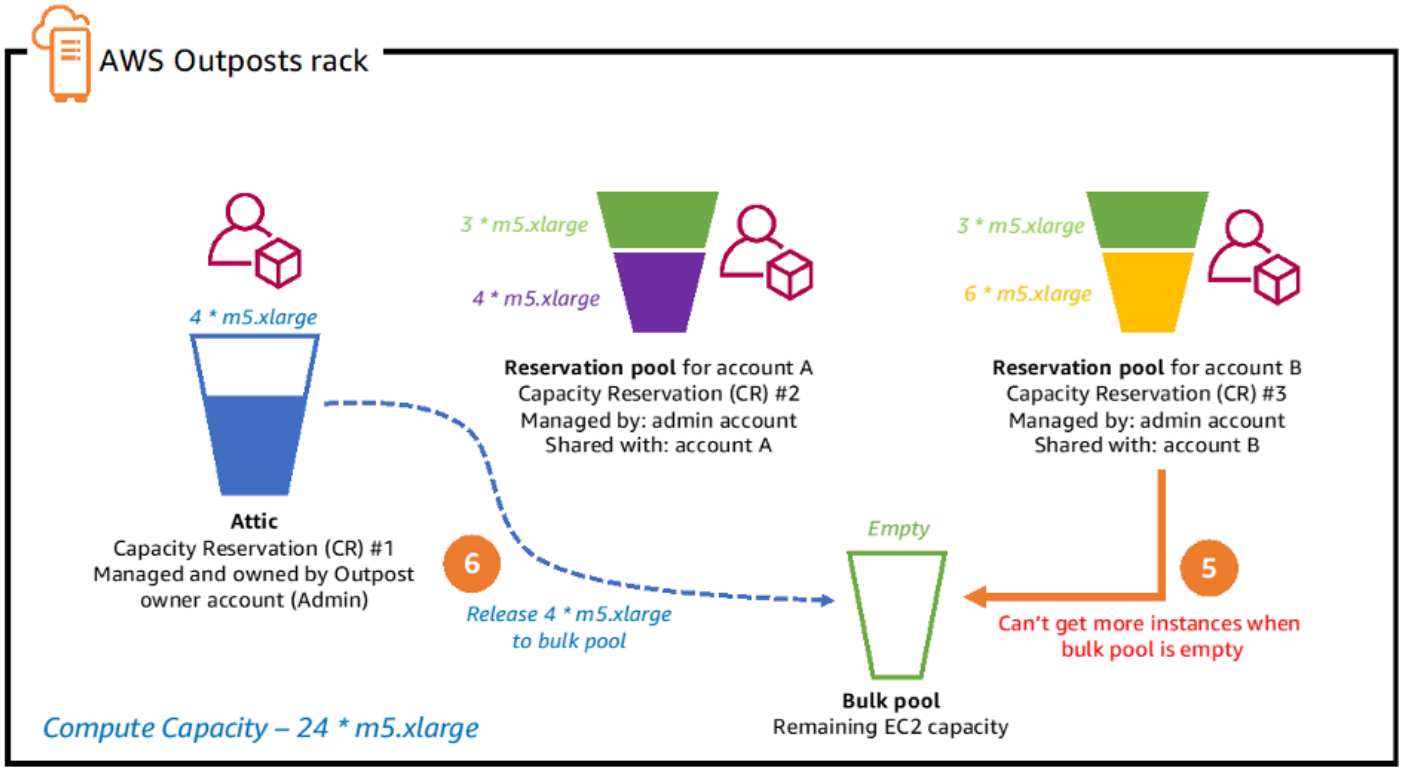


Amazon CloudWatch と AWS Lambda を使用する AWS Outposts キャパシティの管理

キャパシティ予約は、マルチアカウント環境で使用して、1 つのアカウント、または複数のアカウントを含む AWS Organization 単位 (OU) によって使用される Outpost コンピューティング容量を制御できます。Outposts 上の Amazon EC2 のキャパシティ予約、および Amazon Elastic Kubernetes Service (EKS)、Amazon Elastic Container Service (ECS)、Amazon Elastic Map Reduce (EMR) などのサポートされている Outposts AWS のサービスのキャパシティ予約を作成できます。キャパシティ予約は、Outpost 所有者アカウントの AWS Resource Access Manager (AWS RAM) を通じて作成され、アカウントと共有されます。EC2 キャパシティ予約の共有を使用した AWS Outposts ラックのコンピューティングクォータの作成には、キャパシティ管理を目的として Outpost でキャパシティ予約を実装するための実践的なチュートリアルと追加のガイダンスが用意されています。



Capacity Reservation sharing process steps 1-4



Capacity Reservation sharing process steps 5-6

コンピューティングキャパシティ管理の推奨プラクティス

- Auto Scaling グループに EC2 インスタンスを設定するか、インスタンスの自動リカバリを使用して失敗したインスタンスを再起動します。
- Outpost デプロイのキャパシティモニタリングを自動化し、キャパシティアラームの通知と (オプションで) 自動応答を設定します。
- キャパシティ予約を使用して、AWS Organization 内の他のアカウントと共有されるコンピューティングキャパシティをきめ細かく制御できます。

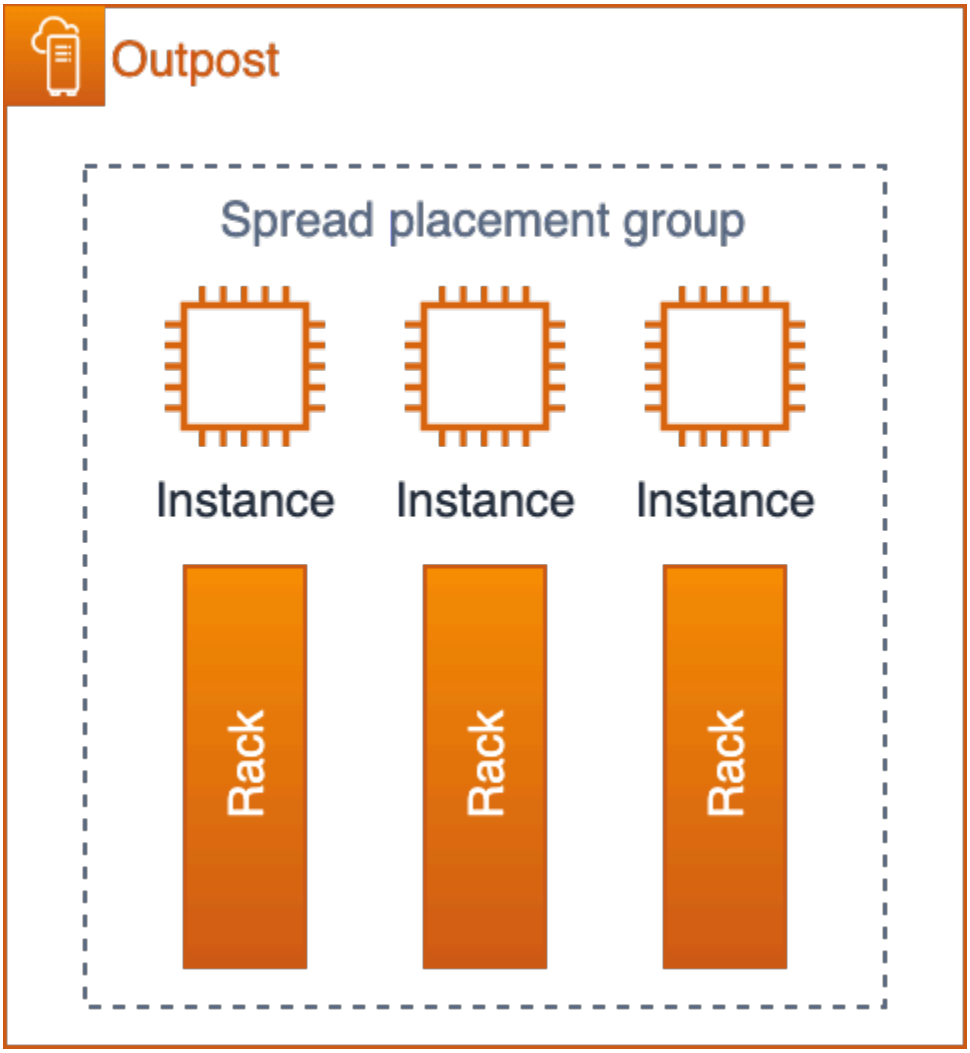
インスタンスのプレイスメント

Outposts のコンピューティングホストの数には限りがあります。アプリケーションが Outposts に複数の関連インスタンスをデプロイする場合、追加の設定を行わなくても、それらのインスタンスは同じホストまたは同じラックのホストにデプロイできます。現在、同じインフラストラクチャで関連するインスタンスを実行するリスクを軽減するために、インスタンスを分散するために使用できるメカニズムは 3 つあります。

マルチ Outpost デプロイ — リージョンのマルチ AZ 戦略と同様に、Outposts を個別のデータセンターにデプロイし、アプリケーションリソースを特定の Outposts にデプロイできます。これにより、目的のOutpost (ラックの論理セット) でインスタンスを実行できます。ダイレクト VPC ルーティングを使用した複数の Outposts 間の VPC 内通信は、Outpost ローカルゲートウェイ (LGW) を使用して Outposts 上のサブネット間にルートを作成し、同じ VPC 内の複数の Outposts にワークロードを分散するために使用できるもう 1 つの戦略です。ラックやデータセンターの障害モードを防ぐためにマルチ Outpost 戦略を採用でき、Outposts が別々の AZ またはリージョンにアンカーされている場合は、AZ またはリージョンの障害モードに対する保護も提供できます。マルチ Outpost アーキテクチャの詳細については、「大規模障害モード」を参照してください。

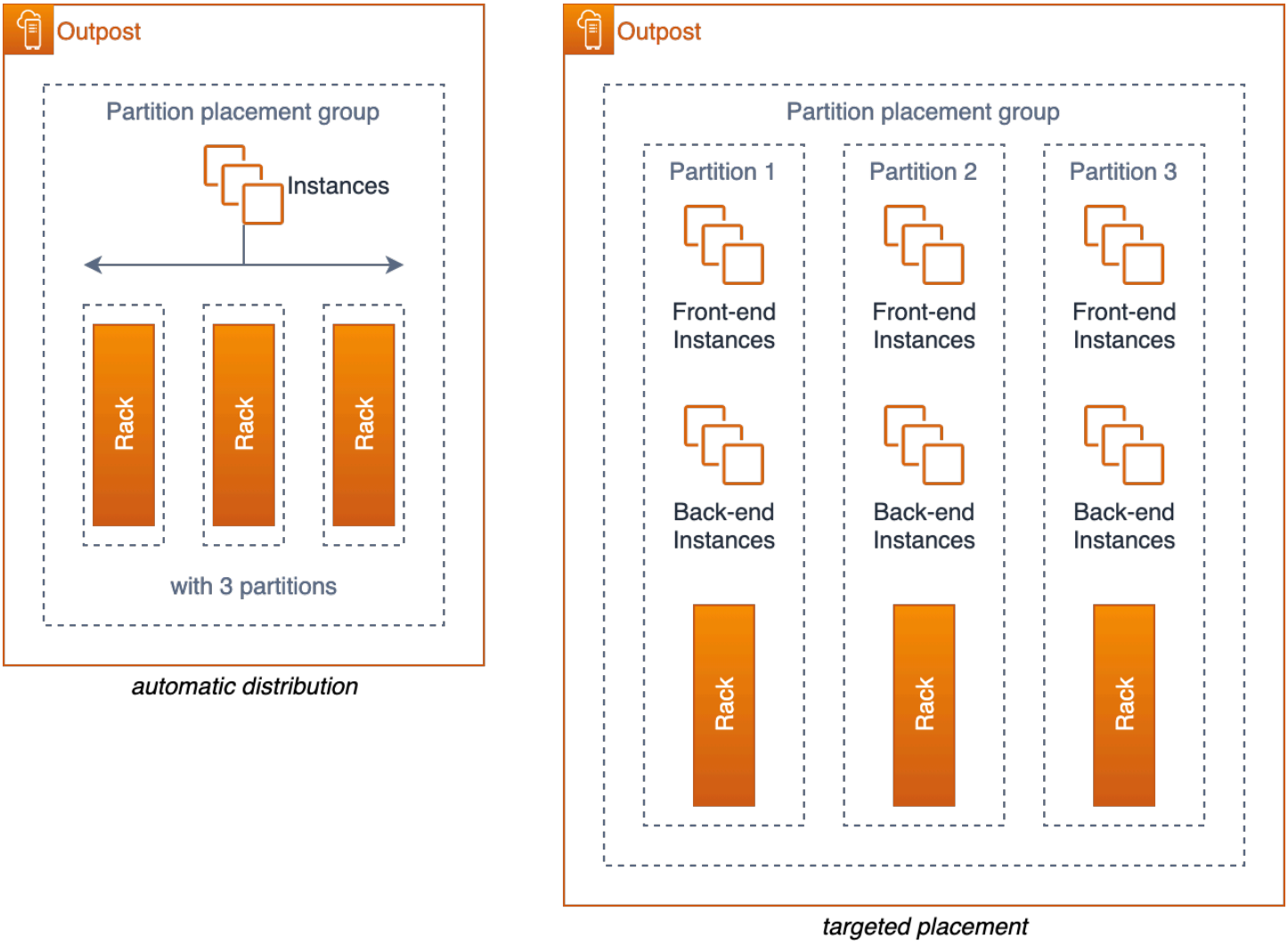
Outposts の Amazon EC2 プレイスメントグループ (シングル Outpost マルチラックインスタンスプレイスメント) – アカウントで作成した Outposts 上にプレイスメントグループを作成できます。これにより、自分のサイトにある Outpost において、基盤となるハードウェア全体でインスタンスを分散できるようになります。Outpost で分散戦略を使用してプレイスメントグループを作成する場合、プレイスメントグループがホストまたはラック全体でインスタンスを分散するように選択できます。

スプレッドプレイスメントグループの使用は、1 つのインスタンスを複数のラックまたはホストに分散させる簡単な方法となり、相互に関連する障害が発生する可能性が低くなります。グループにデプロイできるインスタンスの数は、Outpost にあるホストの数だけです。



3 台のラックがある Outpost の EC2 スプレッドプレイスメントグループ

パーティションプレイスメントグループを使用して、インスタンスを複数のラックに分散することもできます。自動分散を使用して、グループ内のパーティションにインスタンスを分散したり、選択したターゲットパーティションにインスタンスをデプロイしたりできます。インスタンスをターゲットパーティションにデプロイすると、選択したリソースを同じラックにデプロイしながら、他のリソースをラック全体に分散できます。例えば、3 台のラックで構成される論理 Outpost がある場合、3 つのパーティションで構成されるパーティションプレイスメントグループを作成すると、ラック全体にリソースを分散できます。



3 台のラックがある Outpost の EC2 パーティションプレイスメントグループ

クリエイティブサーバースロッティング — シングルラックの Outpost を使用している場合や、Outposts で使用しているサービスがプレイスメントグループをサポートしていない場合は、クリエイティブスロッティングを使用してインスタンスが同じ物理サーバーにデプロイされないようにすることができます。関連するインスタンスが同じ EC2 インスタンスサイズの場合、サーバーをスロットして、各サーバーに設定されるそのサイズのスロットの数を制限して、スロットをサーバー全体に分散できる場合があります。サーバースロッティングは、1 台のサーバーで実行できる (そのサイズの) インスタンスの数を制限します。

例として、先に図 13 に示したスロッティングレイアウトを考えてみましょう。アプリケーションがこのスロッティングレイアウトで設定された Outpost に 3 つの `m5.4xlarge` インスタンスをデプロイする必要がある場合、EC2 は各インスタンスを別々のサーバーに配置し、サーバーで追加の

m5.4xlarge スロットを開くようにスロッティング設定が変更されない限り、これらのインスタンスを同じサーバーで実行される可能性はありません。

コンピューティングインスタンスプレイスメントの推奨プラクティス

- Outposts の Amazon EC2 プレイスメントグループを使用して、1 つの論理 Outpost 内の複数のラックにまたがるインスタンスのプレイスメントを制御できます。
- Outpost を 1 台の中型または大型 Outpost ラックで注文する代わりに、容量を小または中規模の 2 台のラックに分割することを検討してください。そうすれば、複数のラックにインスタンスを分散する EC2 プレイスメントグループの機能を活用できます。
- Outposts の Amazon EC2 プレイスメントグループを使用して、EKS ノードグループ、EKS ローカルクラスターのコントロールプレーンノード、および ECS タスクのプレイスメントに影響を与えることができます。
- VPC 内通信を使用して、同じ VPC 内の複数の Outposts にワークロードを分散します。

ストレージ

AWS Outposts ラックサービスには次の 3 つのストレージタイプがあります。

- サポートされている Amazon EC2 インスタンスタイプのインスタンスストレージ
- 永続的なブロックストレージ用の Amazon Elastic Block Store (EBS) gp2 ボリューム
- ローカルオブジェクトストレージ用の Amazon Simple Storage Service on Outposts (S3 on Outposts)

インスタンスストレージは、サポートされているサーバー (C5d、M5d、R5d、G4dn、I3en) で提供されます。リージョンと同様に、インスタンスストア内のデータは、(実行中の) インスタンスのライフタイムの間のみ保持されます。

Outposts EBS ボリュームと S3 on Outposts オブジェクトストレージは、AWS Outposts ラックマネージドサービスの一環として提供されます。Outpost ストレージプールの容量管理はお客様の責任となります。お客様は Outpost を注文する際に EBS および S3 ストレージのストレージ要件を指定します。AWS は、リクエストされたストレージ容量を提供するのに必要なストレージサーバーの数を Outpost に設定します。AWS は、EBS および S3 on Outposts ストレージサービスの可用性に責任を持ちます。Outpostに可用性の高いストレージサービスを提供するために、十分な数のストレージサーバーがプロビジョニングされています。ストレージサーバーが 1 台故障しても、サービスが中断されたり、データが失われたりすることはないはずです。

AWS Management Console と CloudWatch メトリクスを使用して、Outpost EBS と S3 on Outposts の容量使用率をモニタリングできます。

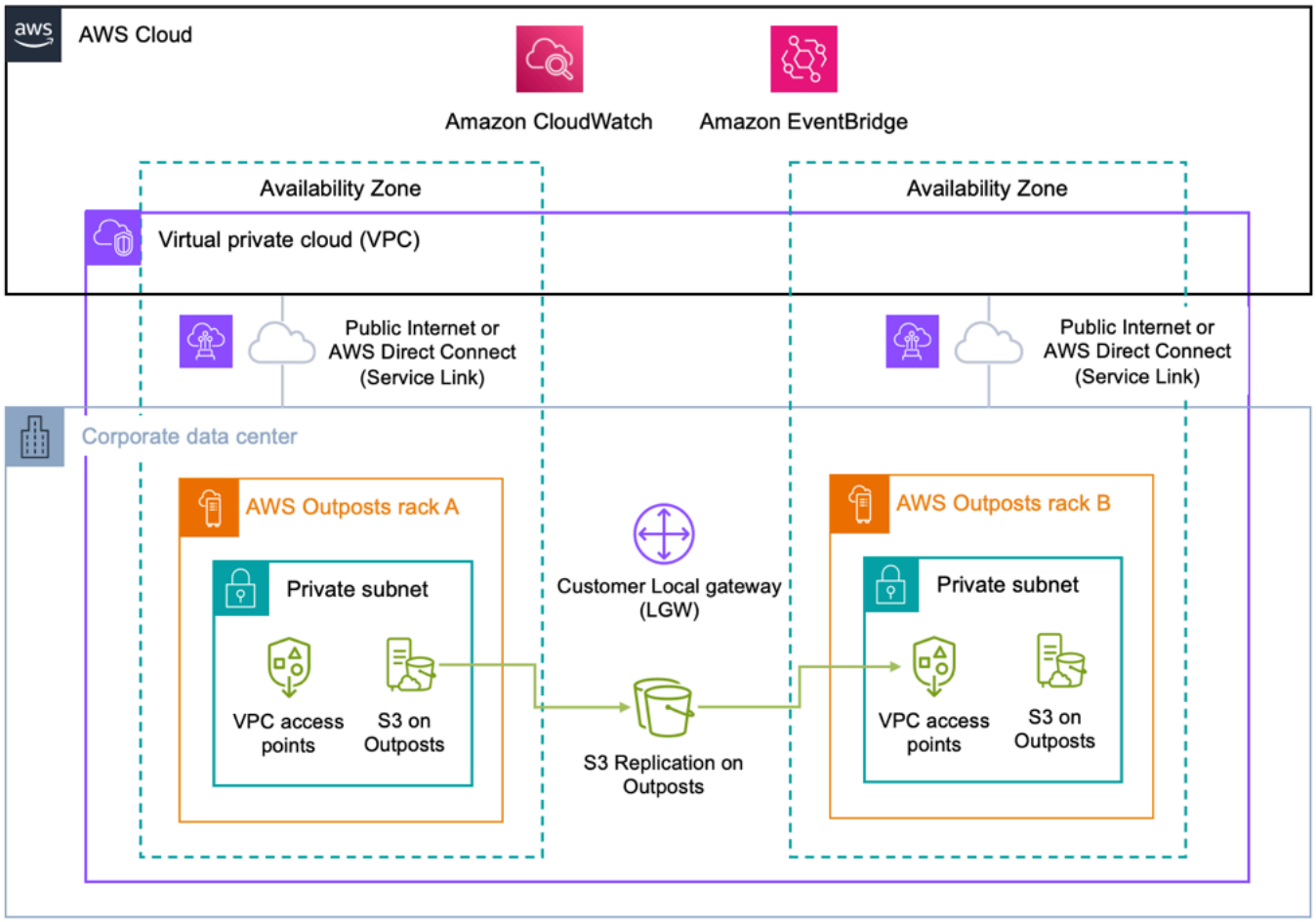
データ保護

EBS ボリュームの場合: AWS Outposts ラックは、EBS ボリュームのスナップショットをサポートし、ブロックストレージのデータを保護するためのシンプルで安全なデータ保護メカニズムを提供します。スナップショットは、EBS ボリュームのポイントインタイムの増分バックアップです。デフォルトでは、Outpost 上のボリュームの Amazon EBS ボリュームのスナップショットは、リージョンにある Amazon S3 に保存されます。Outposts で S3 on Outposts 容量を設定している場合は、EBS Local Snapshots on Outposts を使用し、S3 on Outposts ストレージを使用して、Outposts にスナップショットをローカルにに保存できます。

S3 on Outposts バケットの場合 (データレジデンシーのユースケース):

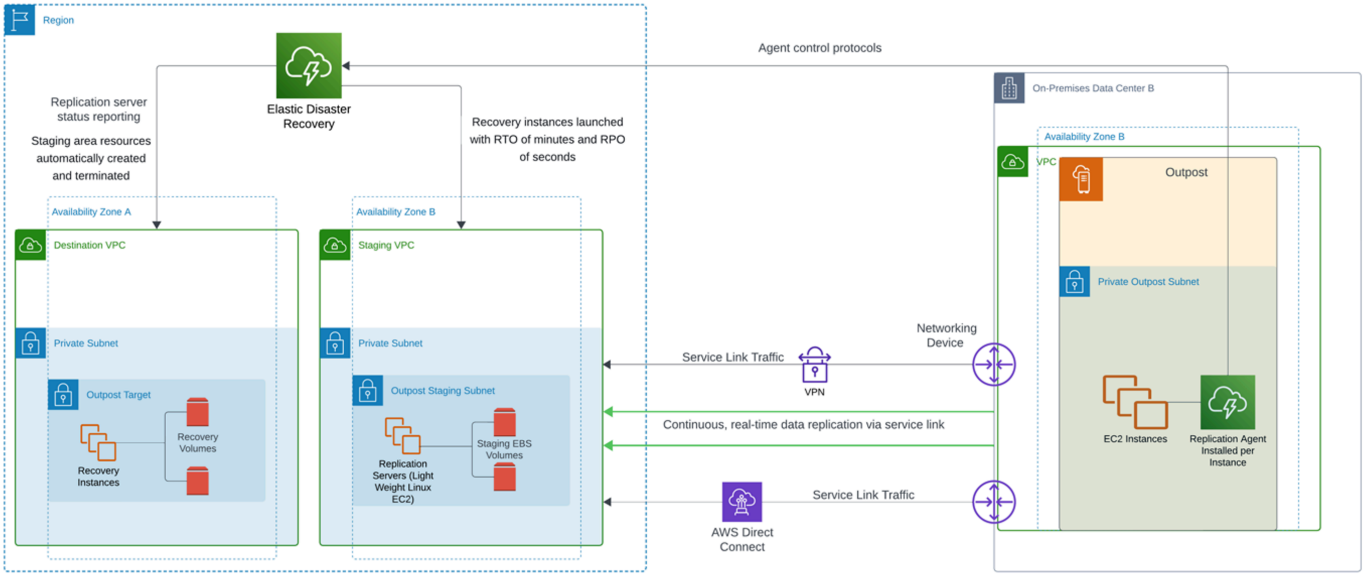
- Outposts の S3 バージョニングを使用して、すべての変更とオブジェクトの履歴を保存できます。S3 バージョニングを有効にすると、オブジェクトの複数の異なるコピーが同じバケット内に保存されます。S3 バージョニングを使用すると、Outposts バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元できます。S3 バージョニングによって、意図しないユーザーアクションやアプリケーション障害から復旧できます。
- S3 Replication on Outposts を使用して、S3 オブジェクトを別の Outpost または同じ Outpost 上の別のバケットに自動的にレプリケートするレプリケーションルールを作成および設定できます。レプリケーション中、S3 on Outposts オブジェクトはお客様のローカルゲートウェイ (LGW) を介して送信され、AWS リージョンに戻ることはありません。S3 Replication on Outposts を使用すると、特定のデータ境界内でデータを自動的にレプリケートして、データの冗長性とコンプライアンスの要件に対応できる、簡単で柔軟な方法が提供されます。

  S3 Replication on Outposts では、オブジェクトレプリケーションのステータスをモニタリングするための詳細なメトリクスと通知も提供されます。Amazon CloudWatch を使用して、保留中のバイト数、保留中のオペレーション、および Outposts のソースバケットとターゲットバケット間のレプリケーションレイテンシーを追跡することで、レプリケーションの進行状況をモニタリングできます。設定の問題をすばやく診断して修正するには、レプリケーションの障害イベントに関する通知を受け取るように Amazon EventBridge を設定することもできます。設定方法の詳細については、YouTube 動画「Amazon S3 Replication on Outposts」を参照してください。



S3 on Outposts バケット (データレジデンシー以外のユースケース) から AWS リージョンへの転送: AWS DataSync を使用して、Outpost とリージョン間の Amazon S3 on Outposts データ転送を自動化できます。DataSync では、転送する内容、転送するタイミング、使用する帯域幅を選択できます。オンプレミスの S3 on Outposts バケットを AWS リージョン内の S3 バケットにバックアップすると、データ耐久性の 99.999999999% (イレブンナイン) と追加のストレージ階層 (標準、低頻度アクセス、Glacier) を活用して、リージョンの S3 サービスで利用できるコストを最適化できます。

インスタンスレプリケーション: AWS Elastic Disaster Recovery (AWS DRS) を使用して、個々のインスタンスと接続されたブロックストレージをオンプレミスシステムから Outpost、Outpost からリージョン、リージョンから Outpost、または 1 つの Outpost から別の Outpost にレプリケートできます。「Architecting for Disaster Recovery on AWS Outposts Racks with AWS Elastic Disaster Recovery」ブログ投稿では、これらの各シナリオと、AWS DRS を使用してソリューションを設計する方法について説明しています。



Outpost からリージョンへのディザスタリカバリ (DR)

AWS Outposts ラックを AWS DRS の宛先 (レプリケーションターゲット) として使用するには、レプリケートされた Amazon EBS スナップショットを保存するために使用される S3 on Outposts ストレージが必要です。またフェイルバックのために、ソース Outposts にも S3 on Outposts ストレージが必要です。AWS DRS を使用するには、Outposts ラックでダイレクト VPC ルーティング (DVR) を使用している必要があります。AWSDRS は Outposts のマネージドサービスインスタンスの保護には使用できません。EC2 インスタンスとそのアタッチされた EBS ボリュームのディザスタリカバリでのみサポートされます。

データ保護の推奨プラクティス:

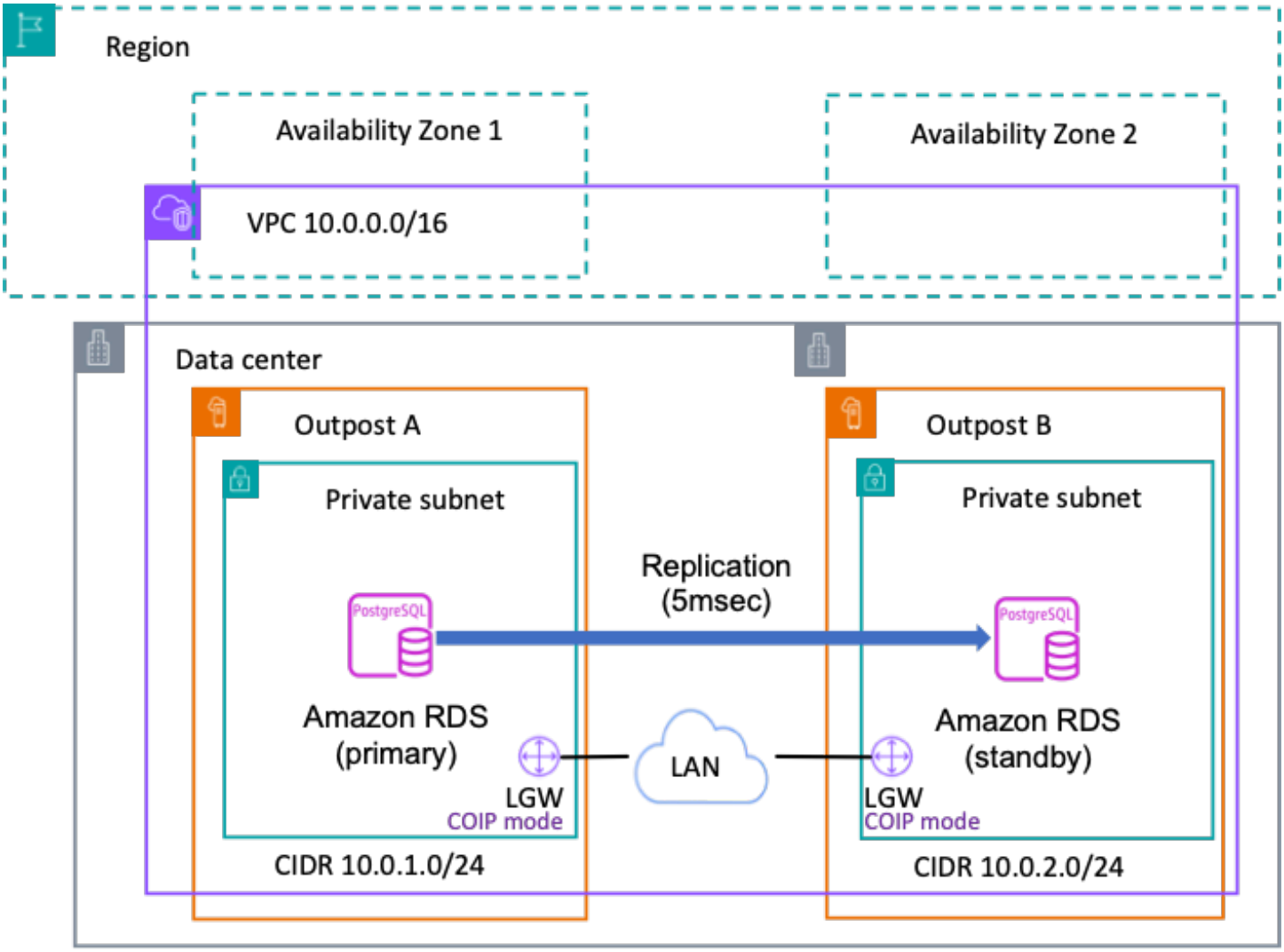
- EBS スナップショットを使用して、リージョンの Amazon S3 または S3 on Outposts へのブロックストレージボリュームのポイントインタイムバックアップを作成します。
- S3 on Outposts オブジェクトのバージョニングを使用して、オブジェクトの複数のバージョンと履歴を管理します。
- S3 Replication on Outposts を使用すると、オブジェクトデータを別の Outpost に自動的にレプリケートできます。
- データレジデンシー以外のユースケースでは、AWS DataSync を使用して、S3 on Outpost に保存されているオブジェクトをリージョンの Amazon S3 にバックアップします。
- AWS DRS を使用して、オンプレミスシステム、論理 Outposts、リージョンの間でインスタンスをレプリケートします。

データベース

AWS Outposts 上の Amazon Relational Database Service (RDS) は、RDS for SQL Server、RDS for MySQL、および RDS for PostgreSQL データベースを AWS Outposts デプロイに拡張します。高可用性アーキテクチャを提供する必要があるデプロイの場合、Amazon RDS は AWS Outposts 上の PostgreSQL および MySQL のマルチ AZ インスタンスデプロイをサポートします。

マルチ AZ を使用した Amazon RDS on Outposts

マルチ AZ 配置では、Amazon RDS はプライマリ DB インスタンスを 1 つの AWS Outposts に作成し、RDS は別の Outpost 上にあるスタンバイ DB インスタンスにデータを同期的にレプリケートします。回復力のあるアーキテクチャを提供するには、2 つの AWS Outposts を特定のリージョン内の異なるアベイラビリティーゾーンに固定し、顧客所有の IP (CoIP) モデルで動作させる必要があります。プライマリインスタンスとスタンバイ間のレプリケーションを許可するには、2 つの Outposts 間にラウンドトリップタイム (RTT) レイテンシーが 1 桁ミリ秒のネットワークリンクが必要です。5 ミリ秒以下をお勧めします。また、レプリケーションジョブのキューイングを避けるために、Outposts 間のレプリケーションリンクを十分な帯域幅で設定することも検討してください。



マルチ AZ を使用した Outposts 上の Amazon RDS

マルチ AZ を使用した Amazon RDS on Outposts に関する考慮事項

マルチ AZ での Amazon RDS on Outposts のデプロイに関する以下の考慮事項を確認してください。

- 同じ AWS リージョン内の異なるアベイラビリティーゾーンに固定された Outposts デプロイが少なくとも 2 つあること。
- プライマリインスタンスとスタンバイインスタンスの両方に、Outposts デプロイごとに 1 つの VPC と 1 つのサブネットが必要です。
- DB インスタンスの VPC をすべてのローカルゲートウェイルートテーブルに関連付けます。
- Outposts が顧客所有の IP ルーティングを使用していることを確認します。

- ローカルネットワークでは、UDP ポート 500 を使用する Outposts for Internet Security Association と Key Management Protocol (ISAKAMP) と、UDP ポート 4500 を使用する IPsec Network Address Translation Traversal (NAT-T) との間のアウトバウンドおよび関連するインバウンドトラフィックを許可する必要があります。
- ローカル RDS バックアップは、マルチ AZ 配置ではサポートされていません。
- ワークロードが業界や地域のデータレジデンシー規制に準拠する必要がある場合は、規制当局に相談して、マルチ AZ RDS が要件を満たしているかどうかを判断してください。

詳細については、「AWS Outposts 上での Amazon RDS のマルチ AZ 配置の使用」を参照してください。

AWS Outposts 上の Amazon RDS リードレプリカ

Amazon RDS リードレプリカは、Amazon RDS データベース (DB) インスタンスのパフォーマンスと耐久性を強化します。これにより、単一 DB インスタンスの容量制約にとらわれることなく伸縮性をもってスケールアウトし、読み取り負荷の高いデータベースワークロードに対応できます。Amazon RDS on AWS Outposts では、MySQL および PostgreSQL の DB エンジンの組み込みのレプリケーション機能を使用して、ソースの DB インスタンスからリードレプリカを作成できます。ソース DB インスタンスがプライマリ DB インスタンスになります。プライマリ DB インスタンスに対して行った更新は、リードレプリカに非同期的にコピーされます。リードレプリカは顧客所有 IP (CoIP) モデルを使用し、レプリケーションはローカルネットワーク上で実行されます。

Amazon RDS on Outposts リードレプリカに関する考慮事項

リードレプリカの Amazon RDS on Outposts デプロイに関する次の考慮事項を確認してください。

- RDS on Outposts DB インスタンスで RDS for SQL Server のリードレプリカを作成することはできません。
- RDS on Outposts では、クロスリージョンリードレプリカはサポートされません。
- RDS on Outposts では、カスケードリードレプリカはサポートされません。
- ソース RDS on Outposts DB インスタンスは、ローカルバックアップを持つことができません。ソース DB インスタンスのバックアップターゲットは、AWS リージョン である必要があります。頻繁に変更されるデータや大量の書き込みトラフィックがある AWS リージョンのデータベースに RDS バックアップを送信するには、少なくとも 500 mbps のサービスリンク接続があることを確認します。
- リードレプリカにはお客様が所有する IP (CoIP) プールが必要です。

- RDS on Outposts のリードレプリカは、ソース DB インスタンスと同じ仮想プライベートクラウド (VPC) でのみ作成できます。
- RDS on Outposts のリードレプリカは、ソース DB インスタンスと同じ VPC 内の同じ Outpost または別の Outpost に配置できます。
- AWS KMS 外部キーストア (XKS) で暗号化された DB インスタンスのリードレプリカを作成することはできません。
- リードレプリカは、ソースのデータベースがマルチ AZ DB インスタンスであるかどうかに関係なく、マルチ AZ DB インスタンスとして作成できます。

AWS Outposts 上の Amazon RDS ストレージのオートスケーリング

ワークロードが予測不能な場合は、Amazon RDS DB インスタンスのストレージの自動スケーリングを有効にすることができます。AWS Outposts 上の Amazon Relational Database Service (Amazon RDS) は、手動および自動ストレージスケーリングをサポートしています。ストレージの自動スケーリングを有効にすると、Amazon RDS は DB インスタンスの空きデータベース容量が不足していることを検出すると、Outposts のデプロイ用にサイズ設定された EBS 容量に基づいてストレージを自動的にスケールアップします。この機能は、「Amazon RDS ストレージの自動スケーリングによる容量の自動管理」にある自動スケーリングに適用される特定の要因があるリージョンで提供される機能と同じ機能を提供します。EBS リソースは Outpost でプロビジョニングされる容量に制限されるため、Outposts の RDS インスタンスに割り当てられる最大ストレージを慎重に管理することが重要です。Amazon RDS ストレージの自動スケーリングを使用すると、最大ストレージ制限を設定して、デプロイが利用可能な EBS 容量内に留まるようにします。Outposts の容量管理の詳細については、このホワイトペーパーの「キャパシティ管理」セクションを参照してください。

Amazon RDS on AWS Outposts のローカルバックアップ

AWS Outposts 上の Amazon RDS ローカルバックアップを使用すると、Outposts にローカルに保存されている S3 から直接 RDS DB インスタンスを復元できます。これにより、データレジデンシー要件を満たし、AWS リージョンからの復旧と比較してレイテンシーを短縮できます。Amazon RDS on AWS Outposts には、次の復元オプションがあります。

- 親リージョンまたは Outposts にローカルに保存されている手動 DB スナップショットから。
- 自動バックアップ (ポイントインタイムリカバリ)。
  - 親 AWS リージョンから復元する場合、バックアップは AWS リージョンまたは Outpost に保存できます。

- Outposts から復元する場合、バックアップは S3 をサポートする Outposts にローカルに保存する必要があります。

AWS Outposts 上の Amazon RDS ローカルバックアップに関する考慮事項

AWS Outposts で Amazon RDS ローカルバックアップを利用するには、以下の考慮事項を参照してください。

- バックアップをローカルに保存するには、S3 on Outposts の容量が必要です。
- ローカルバックアップは MySQL および PostgreSQL DB インスタンスでサポートされています。
- ローカルバックアップは、マルチ AZ インスタンスデプロイまたはリードレプリカではサポートされていません。

RDS on AWS Outposts のスナップショットのエクスポートと復元

スナップショットを S3 にエクスポートし、Amazon S3 から DB インスタンスを復元する: RDS スナップショットは AWS リージョンの Amazon S3 から直接エクスポートまたは復元できますが、これは AWS Outposts 環境内ではサポートされていません。

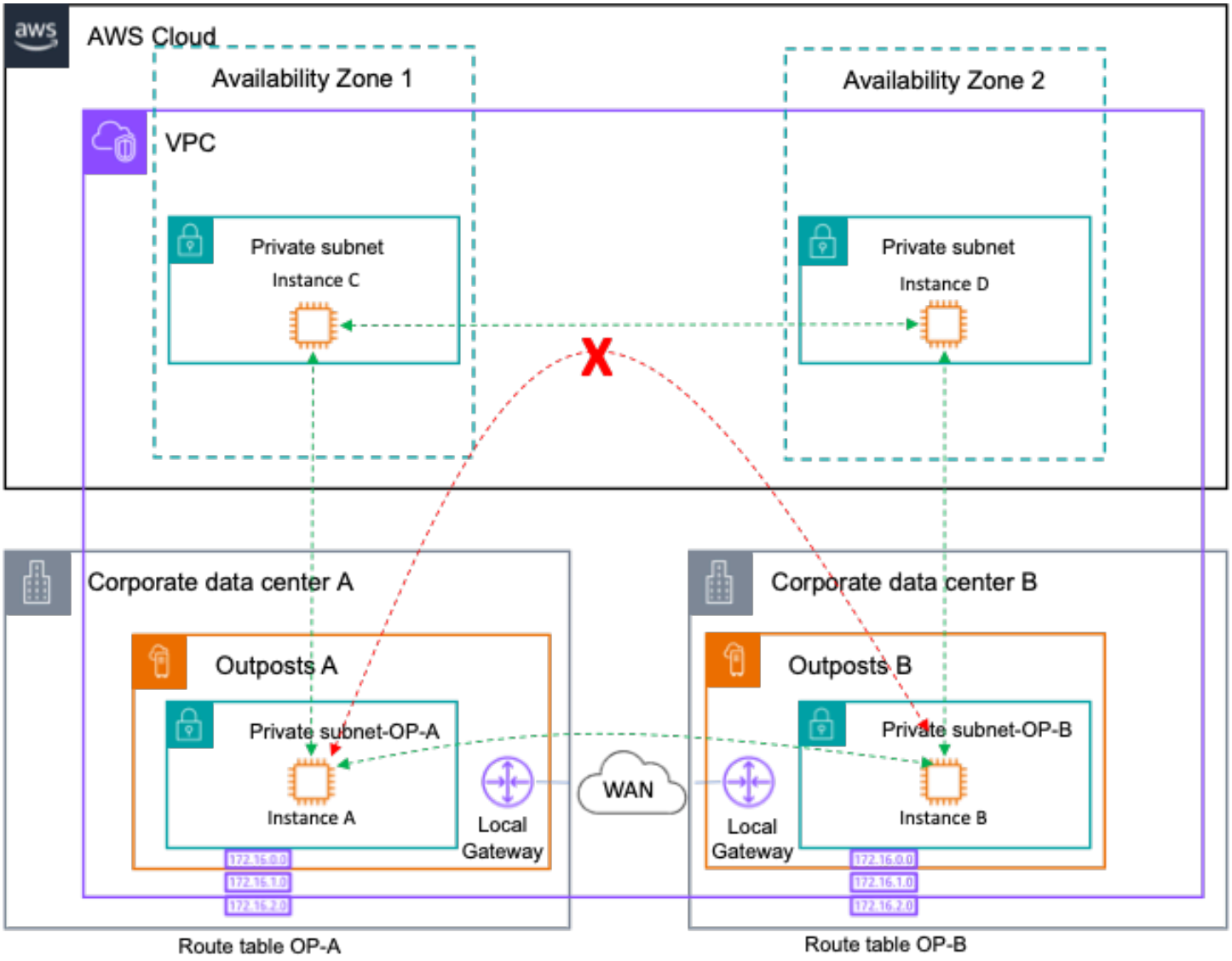
大規模障害モード

ラック、データセンター、アベイラビリティーゾーン (AZ)、またはリージョンの障害などの大規模な障害モードを軽減する HA アーキテクチャを設計するには、独立した電源と WAN 接続を備えた別々のデータセンターに、十分なインフラストラクチャ容量を備えた複数の Outposts をデプロイする必要があります。Outposts は、AWS リージョン または複数のリージョン内のさまざまなアベイラビリティーゾーン (AZ) にアンカーします。また、同期または非同期のデータ複製とワークロードトラフィックのリダイレクトをサポートするために、ロケーション間の耐障害性と十分なサイト間接続をプロビジョニングする必要があります。アプリケーションアーキテクチャに応じて、グローバルに利用可能な Amazon Route 53 DNS と Amazon Route 53 on Outposts を使用して、トラフィックを目的の場所に誘導し、大規模な障害が発生した場合にトラフィックが存続する場所に自動的にリダイレクトされるようにすることができます。

Outposts ラックの VPC 内ルーティング

AWS Outposts ラックは、複数の Outposts 間の VPC 内通信をサポートします。2 つの異なる論理 Outposts のリソースは、Outpost ローカルゲートウェイ (LGW) を使用して、同じ VPC 内のサブ

ネット間でトラフィックをルーティングすることで相互に通信できます。複数の Outposts 間の VPC 内通信では、ローカル LGW をネクストホップとして使用して、他の Outposts サブネットにより具体的なルートを追加することで、Outposts サブネットに関連付けられたルートテーブルのローカルルートを上書きできます。これにより、2 つの Outposts ラック間の Amazon ECS や AWS Outposts 間の Amazon EKS クラスターなど、2 つの論理 Outposts 間で VPC をまたぐ必要があるアプリケーションの設計にメリットがあります。

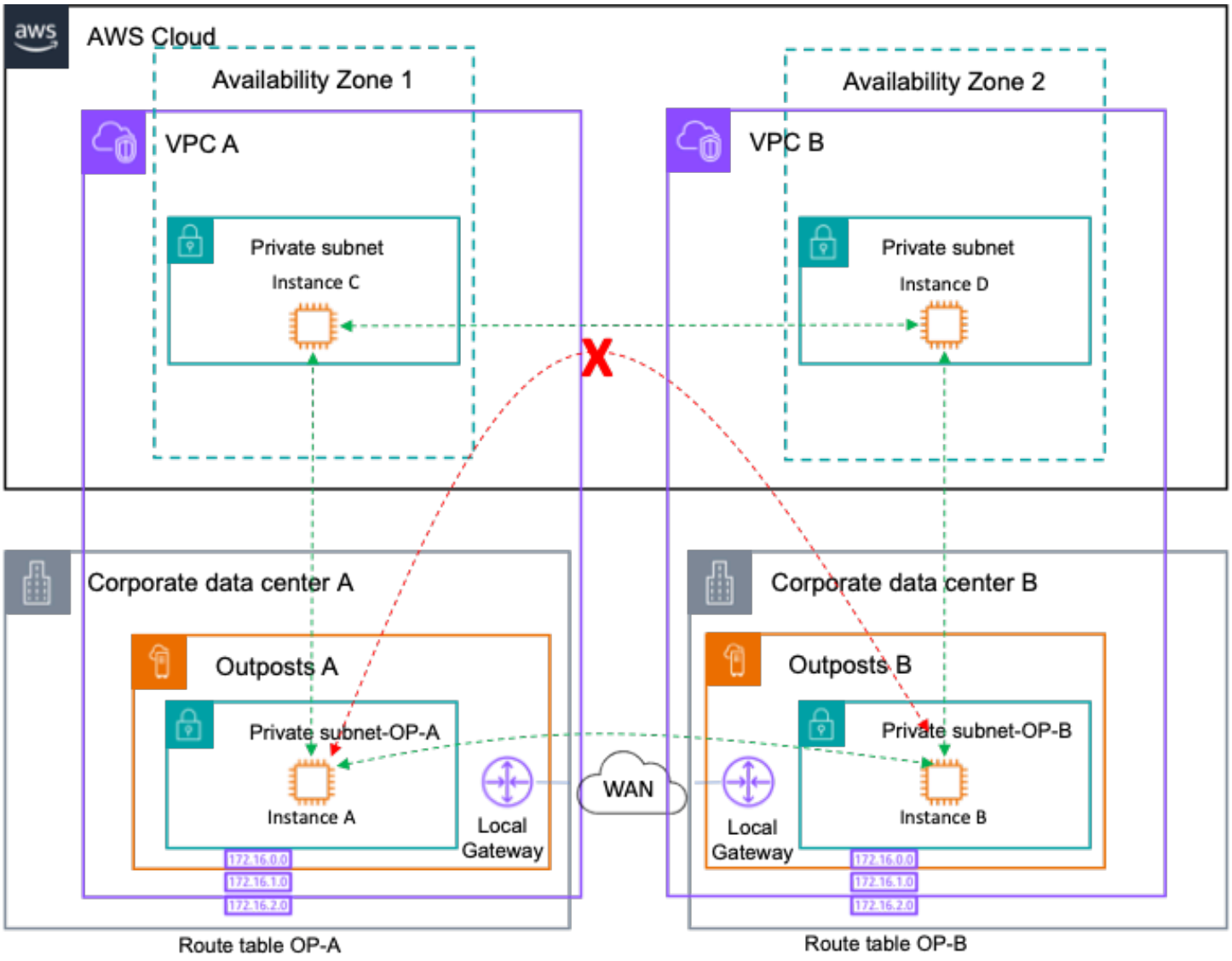


複数の論理 Outposts を持つ単一の VPC のネットワークパス

リージョンを経由する Outpost から Outpost へのトラフィックは、アンチパターンであるためブロックされます。このようなトラフィックは両方向で出力料金が発生し、お客様の WAN を介してトラフィックをルーティングするよりもはるかにレイテンシーが大きくなる可能性があります。

Outposts ラックの VPC 間ルーティング

異なる VPC にデプロイされた 2 つの別々の Outpost 上のリソースは、カスタマーネットワークを介して相互に通信できます。このアーキテクチャをデプロイすると、ローカルのオンプレミスネットワークと WAN ネットワークによって Outposts 間のトラフィックをルーティングし、対応する Outposts/VPC サブネットへのルートを追加できます。



複数の論理 Outposts を持つ複数の VPC のネットワークパス

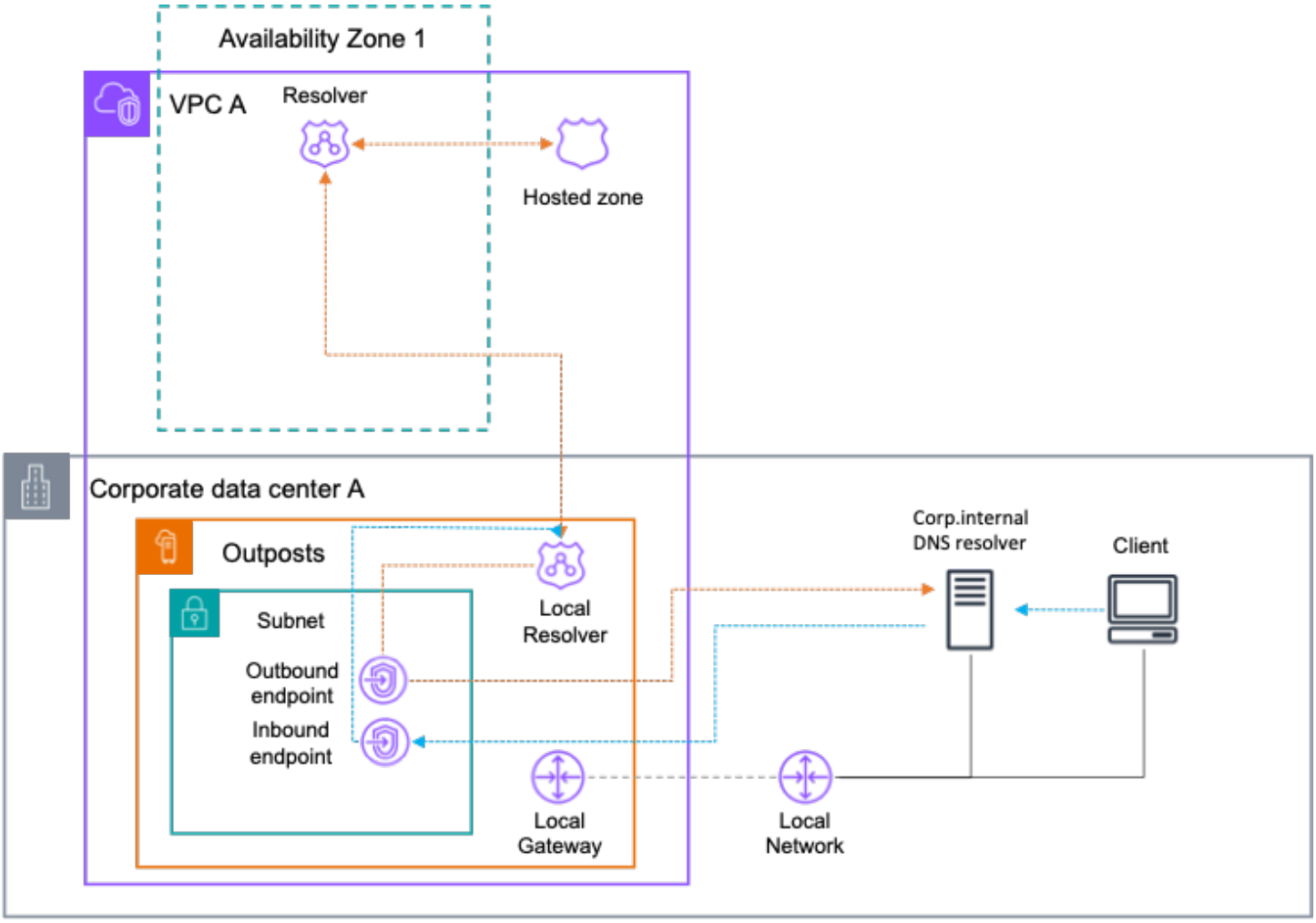
大規模な障害モードからの保護に関する推奨プラクティス:

- 複数の AZ とリージョンにアンカーされた複数の Outpost をデプロイします。
- マルチ Outpost デプロイでは、Outpost ごとに別々の VPC を使用してください。

Outposts 上の Route 53 ローカルリゾルバー

AWS Outposts サービスリンクが一時的な切断の影響を受けると、ローカル DNS 解決が失敗するため、同じ Outposts ラックで実行されている場合でも、アプリケーションやサービスが他のサービスを検出することが困難になります。ただし、AWS Outposts 上の Route 53 リゾルバーを使用すると、親 AWS リージョンへの接続が失われた場合でも、アプリケーションとサービスは引き続きローカル DNS 解決を利用して他のサービスを検出できます。同時に、オンプレミスのホスト名の DNS 解決では、Outposts 上の Route 53 リゾルバーは、クエリ結果がキャッシュされてローカルで提供されるため、レイテンシーの削減に役立ちます。また、Route 53 リゾルバーのエンドポイントと完全に統合されています。

Route 53 リゾルバーのインバウンドエンドポイントは、VPC の外部から受信した DNS クエリを Outposts で実行されているリゾルバーに転送します。対照的に、Route 53 リゾルバーアウトバウンドでは、次の図に示すように、Route 53 リゾルバーがオンプレミスネットワークで管理する DNS リゾルバーに DNS クエリを転送できるようになります。



Outposts 上の Route 53 リゾルバー

Outposts 上の Route 53 リゾルバーに関する考慮事項

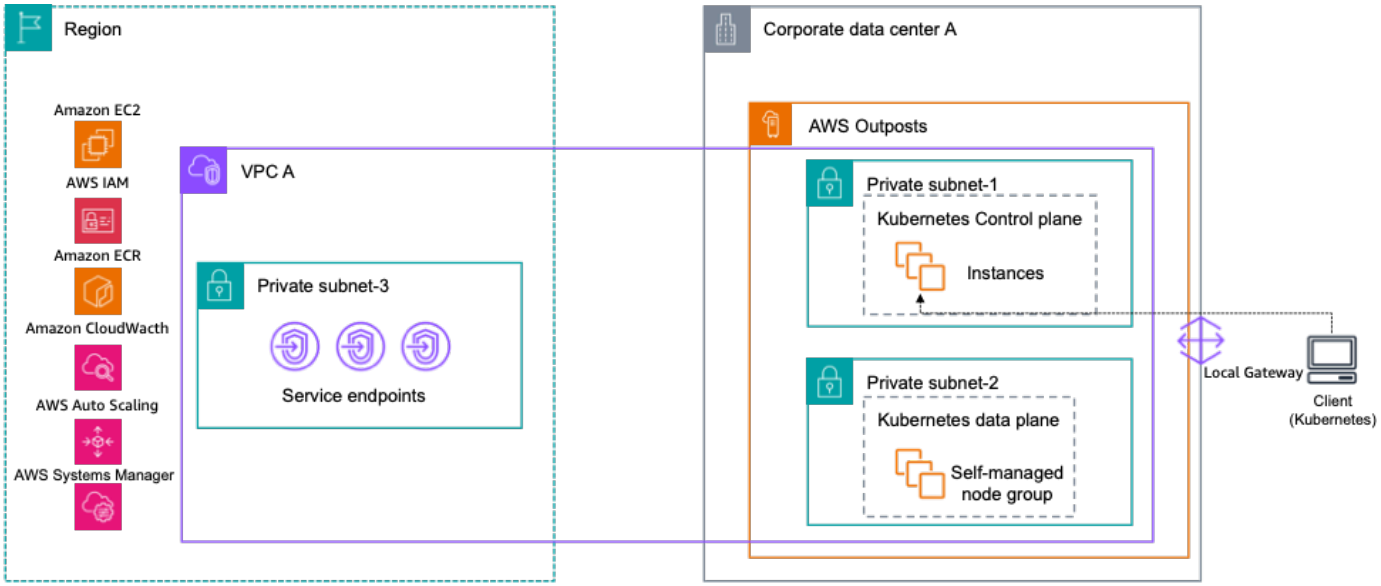
以下の点を考慮します。

- Outposts 上で Route 53 リゾルバーを有効にする必要があり、これは単一の Outposts ID で複数のコンピューティングラックがある場合でも、Outposts のデプロイ全体に適用されます。
- この機能を有効にするには、Outposts に c5.xlarge、m5.large、または m5.xlarge の少なくとも 4 つの EC2 インスタンスの形式でローカルリゾルバーをデプロイするのに十分なコンピューティングキャパシティが必要です。
- プライベート DNS を使用している場合は、Outposts 上の Route 53 リゾルバーでレコードをローカルにキャッシュするために、必要な Outposts VPC とプライベートホストゾーンを共有する必要があります。
- オンプレミス DNS とインバウンドおよびアウトバウンドエンドポイントとの統合を有効にするには、Outposts に Route53 エンドポイントごとに 2 つの EC2 インスタンスをデプロイするのに十分なコンピューティングキャパシティが必要です。

Outposts 上の EKS ローカルクラスター

親リージョンから Outposts サービスリンクが切断されると、コントロールプレーンがリージョンに存在する EKS 拡張クラスターなどのサービスで問題が発生する可能性があります。問題の 1 つは、EKS コントロールプレーンとワーカーノードおよび POD 間の通信が失われることです。ワーカーノードと POD は両方とも、Outposts にローカルに存在するアプリケーションを引き続き動作および処理できますが、Kubernetes コントロールプレーンはそれらを異常と見なし、コントロールプレーンへの接続が回復したときにそれらの交換をスケジュールする場合があります。これにより、接続が回復したときにアプリケーションのダウンタイムが発生する可能性があります。

これを簡素化するために、Outposts 上で EKS クラスター全体をホストするオプションがあります。この設定では、Kubernetes コントロールプレーンとワーカーノードの両方がオンプレミスの Outposts コンピューティングキャパシティでローカルに実行されます。これにより、サービスリンク接続が一時的に切断されても、クラスターは動作し続けます。



Outposts 上の Amazon EKS ローカルクラスター

Outposts 上の EKS ローカルクラスターに関する考慮事項

EKS ローカルクラスターが Outposts にデプロイされる際には、いくつかの考慮事項があります。

- 切断中は、AWS 親リージョンへの EC2 および ASG API コールに依存している限り、新しいワーカーノードの追加やノードグループの自動スケーリングを必要とするクラスター自体の変更を実行するオプションはありません。
- • eksctl AWS Outposts サポートにリストされているローカルクラスターには、サポートされていない一連の機能があります。

結論

AWS Outposts ラックを使用すると、Amazon EC2、Amazon EBS、Amazon S3 on Outposts、Amazon ECS、Amazon EKS、Amazon RDS などの使い慣れた AWS のツールやサービスを使用して、可用性の高いオンプレミスアプリケーションを構築、管理、スケールすることができます。ワークロードは、ローカルで実行、クライアントにサービスを提供、オンプレミスネットワークでアプリケーションやシステムにアクセス、および AWS リージョン 内のすべてのサービスにアクセスできます。Outposts ラックは、オンプレミスシステム、ローカルデータ処理、データレジデンシー、ローカルシステムの相互依存性によるアプリケーション移行で低レイテンシーアクセスを必要とするワークロードに最適です。

Outpost デプロイに、十分な電力、スペース、冷却、および AWS リージョン への耐障害性を備えた接続を提供すれば、可用性の高い単一データセンターサービスを構築できます。また、可用性と耐障害性のレベルを高めるために、複数の Outposts をデプロイし、アプリケーションを論理的および地理的な境界を越えて分散させることもできます。

Outposts ラックを利用すると、オンプレミスのコンピューティング、ストレージ、およびアプリケーションネットワークプールを構築するという、画一的な負荷の大きい作業から解放され、AWS グローバルインフラストラクチャのリーチをデータセンターやコロケーション施設にまで広げることができます。これで、アプリケーションのモダナイゼーション、アプリケーションデプロイの合理化、IT サービスのビジネスインパクトの向上に時間と労力を注ぐことができます。

寄稿者

本ドキュメントの寄稿者は次のとおりです。

- アマゾン ウェブ サービス、通信、プリンシパルソリューションアーキテクト、Jesus Federico
- アマゾン ウェブ サービス、S3 on Outposts、Mallory Gershenfeld
- アマゾン ウェブ サービス、ハイブリッドクラウド、シニアソリューションアーキテクト、Rob Goodwin
- アマゾン ウェブ サービス、AWS Outposts、シニアスペシャリストソリューションズアーキテクト、Chris Lunsford
- アマゾン ウェブ サービス、AWS Outposts、リードアーキテクト、Rohan Mathews
- アマゾン ウェブ サービス、ハイブリッドエッジスペシャリストソリューションアーキテクト、Brianna Rosentrater
- アマゾン ウェブ サービス、プリンシパルハイブリッドエッジスペシャリストソリューションアーキテクト、Leonardo Solano
-

ドキュメント履歴

このホワイトペーパーの更新に関する通知を受け取るには、RSS フィードにサブスクライブしてください。

変更	説明	日付
メジャーな更新	ネットワーク、DRS サポート、Amazon EKS ローカルクラスター、プレイスメントグループ、AWS Outposts 上の Amazon RDS に関する更新を追加	2024 年 11 月 24 日
マイナーな更新	追加のスロッティングガイダンスがキャパシティプランニングに追加されました。	2024 年 2 月 9 日
マイナーな更新	初版発行以降にリリースされた機能を反映するように更新しました。	2023 年 7 月 19 日
マイナーな更新	可用性の高いネットワーク接続の推奨プラクティスを更新しました。	2023 年 6 月 29 日
初版発行	ホワイトペーパーの初回発行。	2021 年 8 月 12 日

> **Note**
>
> RSS の更新を購読するには、使用しているブラウザで RSS プラグインを有効にする必要があります。

注意

お客様は、本書に記載されている情報を独自に評価する責任を負うものとします。本書は、(a) 情報提供のみを目的とし、(b) AWS の現行製品と慣行について説明しており、これらは予告なしに変更されることがあり、(c) AWS およびその関連会社、サプライヤー、またはライセンサーからの契約上の義務や保証をもたらすものではありません。AWS の製品やサービスは、明示または黙示を問わず、一切の保証、表明、条件なしに「現状のまま」提供されます。お客様に対する AWS の責任は AWS 契約によって規定されます。本書は、AWS とお客様との間で締結されるいかなる契約の一部でもなく、その内容を修正するものでもありません。

AWS 用語集

AWS の最新の用語については、「AWS の用語集リファレンス」の「AWS 用語集」を参照してください。