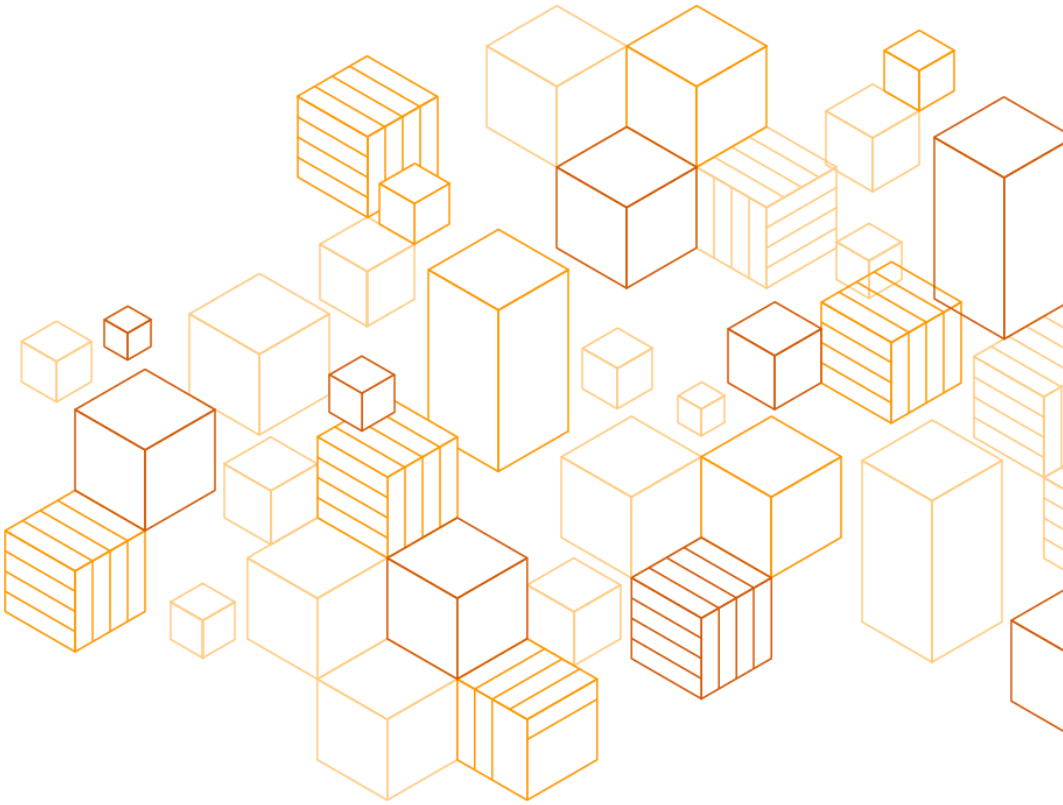
AWS による JPX arrownet 接続

実装ガイド

2020 年 5 月



注意

お客様は、この文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ており、(b) AWS の現行製品とプラクティスを表したものであり、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義務や確約を意味するものではありません。AWS の製品やサービスは、明示または暗示を問わず、いかなる保証、表明、条件を伴うことなく「現状のまま」提供されます。お客様に対する AWS の責任は、AWS 契約により規定されます。本書は、AWS とお客様の間で締結される一切の契約の一部ではなく、その内容を修正することもありません。

目次

概要

本実装ガイドは、アマゾンウェブサービス (AWS) を利用して、株式会社日本取引所グループ(JPX) が運営する arrownet(arrownet version 2) への接続方式に重点を置いて解説しています。本ガイドでは、arrownet と接続するためのシナリオ例とリファレンスアーキテクチャ図を示します。また本ガイドでは、AWS Well-Architected フレームワークの用語と定義を使用して説明しています。

本ガイドは、ネットワーキング、オペレーティングシステム、および運用管理の基本的な概念を理解している IT 領域での意思決定者、およびインフラストラクチャ / ネットワーキングの専門家を対象としています。

JPX arrownet 接続のための AWS クラウドアーキテクチャ

このセクションでは、AWS クラウドを使用して JPX arrownet との接続を確立するために必要なアーキテクチャについて説明します。

この実装ガイドでは、シナリオに基づきリファレンスアーキテクチャについて説明します。用途に適したシナリオを選択ください。また AWS クラウドのセキュリティと効率の利点を最大化するためには、構築フェーズ（要件の定義、設計、構築、テスト）を通じて Well-Architected フレームワークに従っていることをご確認ください。

また本ガイドを利用する前に、次の留意事項 を確認ください。詳細については、JPX arrownet ガイドラインを参照ください。

- AWS とのアクセスポイントを利用する前に、JPX がサポートするサービスを確認してください。
- AWS アクセスポイント を使用する場合、JPX では、クラウド接続に加えて arrownet バージョン 2.0 回線を使用して冗長構成を設定することが推奨されています。本ガイドでは、AWS によるネットワークアーキテクチャについて説明しており、このアーキテクチャだけでは、JPX が推奨する冗長構成を達成できません。したがって、arrownet が提供する既存の接続サービス（arrownet バージョン 2.0 回線）を考慮しながら、arrownet へのネットワーク接続を設計する必要があります。これにより、JPX が推奨する可用性が保証されます。
- arrownet バージョン 2.0 における AWS 接続では、ユニキャストのみを使用できます。マルチキャストは使用できません。

- arrownet バージョン 2.0 における AWS 接続は、JPX プライマリサイトのみ使用できます。サイト切替え発生時では使用できません。

arrownet への接続シナリオ

AWS クラウド は、AWS Direct Connect プライベート仮想インターフェイス（プライベート VIF）を介して JPX arrownet に接続します。

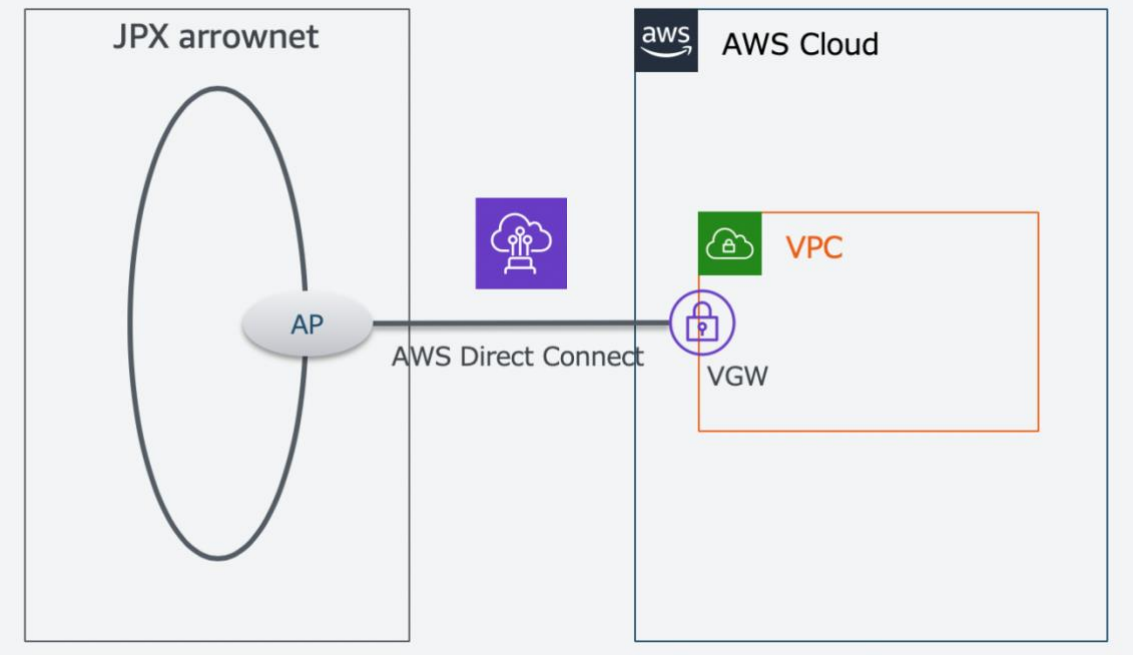


図 *1 - arrownet と AWS* クラウドとの接続の基本

プライベート VIF を介して arrownet への接続許可については、JPX に申請する必要があります。申請手順の詳細については、JPX arrownet version2.0 のガイドラインを参照してください。

次の表に、接続シナリオを示します。

表 *1* - 接続シナリオ

シナリオ区分	概要	詳細説明
シナリオ 1	AWS クラウドに閉じてトラフィックを処理する	市場参加者が利用できる JPX からのトラフィックを AWS 内で 制限して処理します。このシナリオは、ワークロードを AWS に移行した市場参加者、および AWS でワークロードを新たに実装する市場参加者（Fintech スタートアップ など）向けに設計されています。
シナリオ 2	オンプレミスのデータセンターへトラフィックを転送する	市場参加者が使用するオンプレミスのデータセンターにトラフィックを 転送し、JPX からのトラフィックを処理できるようにします。このシナリオは、AWS をトランジットセンターとして使用し、JPX と通信するワークロードをオンプレミスのデータセンターに保持する市場参加者向けに設計されています。
シナリオ 3	ログの集約と分析のアーキテクチャ	JPX と AWS クラウド間のトラフィックから生成されたログを集約し、分析するアーキテクチャ。

シナリオ 1. AWS クラウドに閉じてトラフィックを処理する

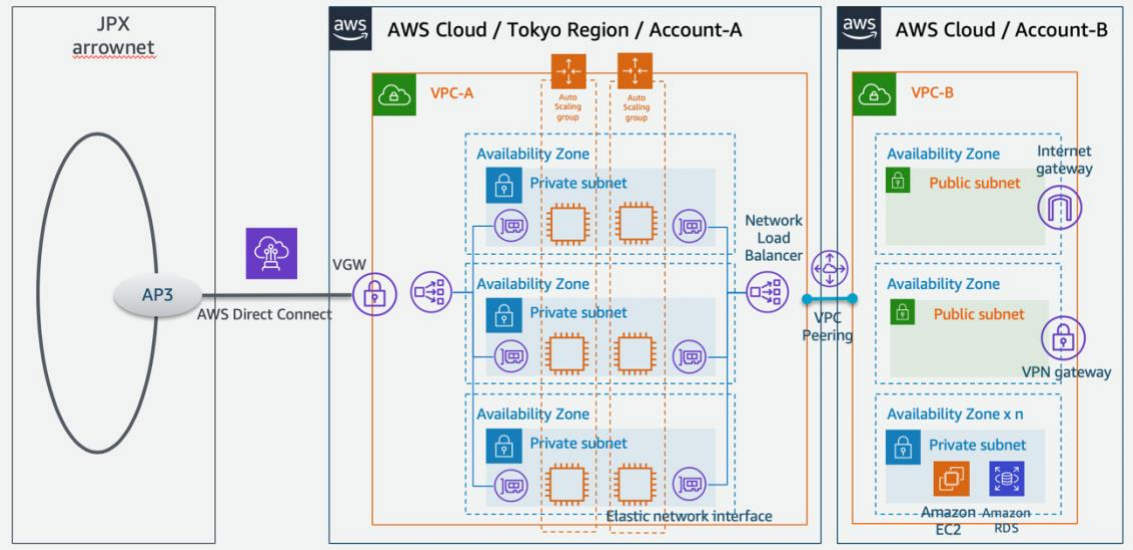


図 2 - AWS クラウドに閉じてトラフィックを処理するためのリファレンスアーキテクチャ

設計の原則

市場参加者には、JPX とやり取りする必要がない内部通信が arrownet に流れることを防ぐ、アーキテクチャが求められます。

- Virtual Private Cloud(VPC) は VPC-A と VPC-B に分かれています。このアーキテクチャは、VPC-A が JPX とのトラフィック処理専用であり、市場参加者のワークロードのリソースが VPC-B に割り当てられるように設計されています。

- 予期せぬトラフィックが市場参加者の VPC-B から VPC-A に流れるような場合には、VPC-A のプロキシサーバー機能（またはセキュリティグループの機能）により、トラフィックが arrownet に流れるのを防ぐことができます。逆に、VPC-A は、arrownet 側からの予期しないトラフィックを防止することもできます。
- 市場参加者が使用するインターネットゲートウェイまたは VPN ゲートウェイを VPC-A に配置 しないでください。VPC-A には、arrownet によって提供される AWS Direct Connect のプライベート VIF が含まれています。
- 市場参加者が所有する VPC-B と通信する場合、VPC-A と VPC-B の間の VPC ピアリング、AWS Transit Gateway、または AWS PrivateLink を使用できます。

アーキテクチャの説明

> **重要：**このシナリオでは、AWS 東京リージョンを使用する必要があります。AWS 東京リージョンの使用は、金融商品取引法で定義されているコンプライアンス要件です。

以下のステップでは、このシナリオのリファレンスアーキテクチャについて説明します。

1. 東京リージョンで VPC-A をセットアップし、AWS Direct Connect の VIF を仮想プライベートゲートウェイ (VGW) にアタッチします。

2. Network Load Balancer(NLB) をデプロイし、Amazon Elastic Compute Cloud(EC2) 仮想サーバーを使用してプロキシサーバーを作成し、ap-northeast-1a、ap-northeast-1c、ap-northeast-1d の 3 つのアベイラビリティーゾーン (AZ) に配置します。
3. Auto Scaling の設定を [Max]:3、[Min]:3、[Desirable]:3 に調整します。JPX の情報系サービスへ接続し、プロトコルが http または https に制限されている場合、NLB の代わりに Application Load Balancer(ALB) を使用できます。さらに、NLB または ALB にアクセスする市場参加者のサーバーまたはブラウザは、DNS サーバーを介して名前を解決できるように設定する必要があります。

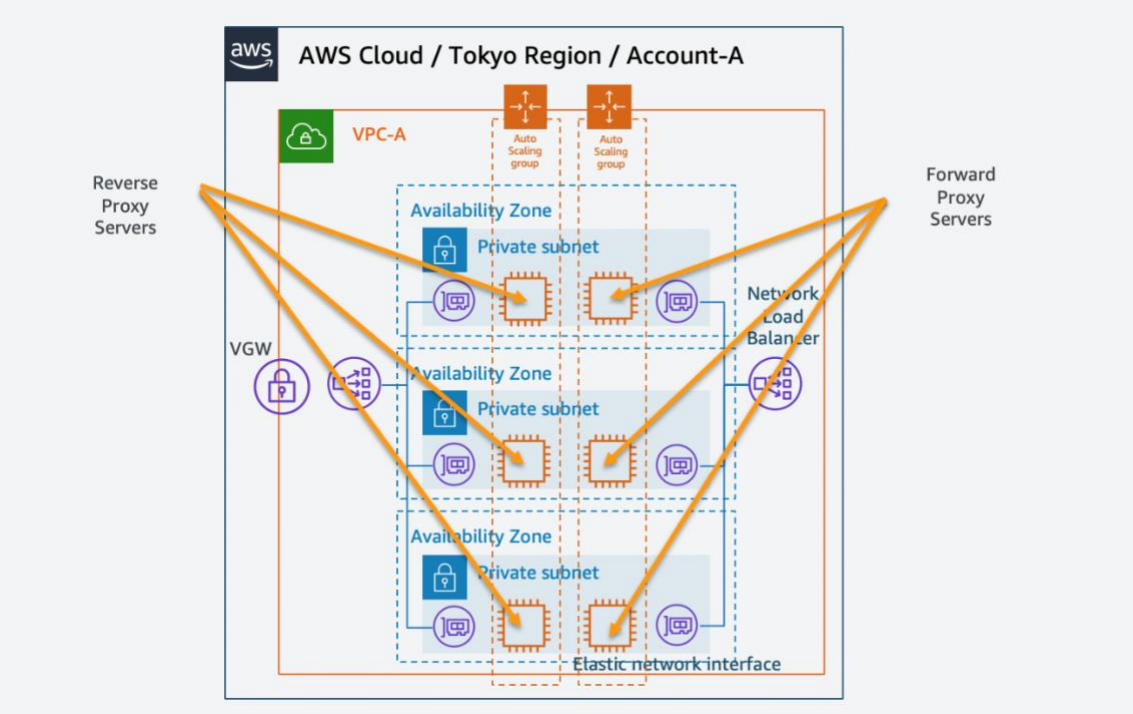


図 3 - プロキシサーバー設定

EC2 設定に関する以下の詳細情報 を参照してください。

1. ネットワーク帯域幅はインスタンスタイプによって変化するため、JPX とのトラフィック量を測定しながら最適なインスタンスタイプを選択します。インスタンスタイプによっては、ネットワーク帯域幅が明確に定義されていない場合があります。インスタンスタイプを選択するときは、この点を念頭に置いてください。
2. T2/T3 インスタンスタイプを使用する場合は、インスタンスの起動時に T2/T3 Unlimited オプションを選択します。これは、CPU クレジットの枯渇によるトラフィックエラーを回避するための設定です。
3. EC2 の動作ステータスをモニタリングするには、 「Amazon CloudWatch 詳細モニタリングを有効にする」オプションを選択することをお勧めします。

4. 本ガイドでは、3AZ 構成のマルチ AZ 設定を使用することをお勧めしていますが、最低でも 2 つのアベイラビリティーゾーン (AZ) の使用をお勧めします。Well Architected フレームワークに基づく信頼性の観点からは、1 アベイラビリティーゾーン (AZ) 設定（シングル AZ 設定）はお勧めしません。Auto Scaling は、Auto Healing と呼ばれるインスタンス障害時の自動復旧の目的で使用され、この設定では常に 2~3 個の EC2 インスタンスの起動状態が維持されます。EC2 をプロキシサーバーにするには、各市場参加者のセキュリティポリシーに準拠した最適なソフトウェアをインストールします。
   NLB+EC2+Auto Scaling の組み合わせには 2 つのセットがあります。1 つは JPX へのトラフィックのためのフォワードプロキシで、もう 1 つは JPX からのトラフィックのためのリバースプロキシです。これらのプロキシは EC2 で設定されており、arrownet との通信を維持するために確実に起動する必要があります。そのため、AZ でリザーブドインスタンス (RI) を採用します。RI を採用すると、コスト効率も向上します。市場参加者は RI を購入して、独自の投資計画を満たすことができます。一方、証券市場ではトラフィックの変動が大きくなるため、初期構築段階で複数のインスタンスタイプをテストして、インスタンスタイプの変更に備えることをお勧めします。

5. VPC-B を 東京リージョン に設定し、VPC-A との通信を許可します。市場参加者のワークロードを処理するために必要なリソースをデプロイします。

シナリオ 2. オンプレミスのデータセンターへトラフィックを転送する

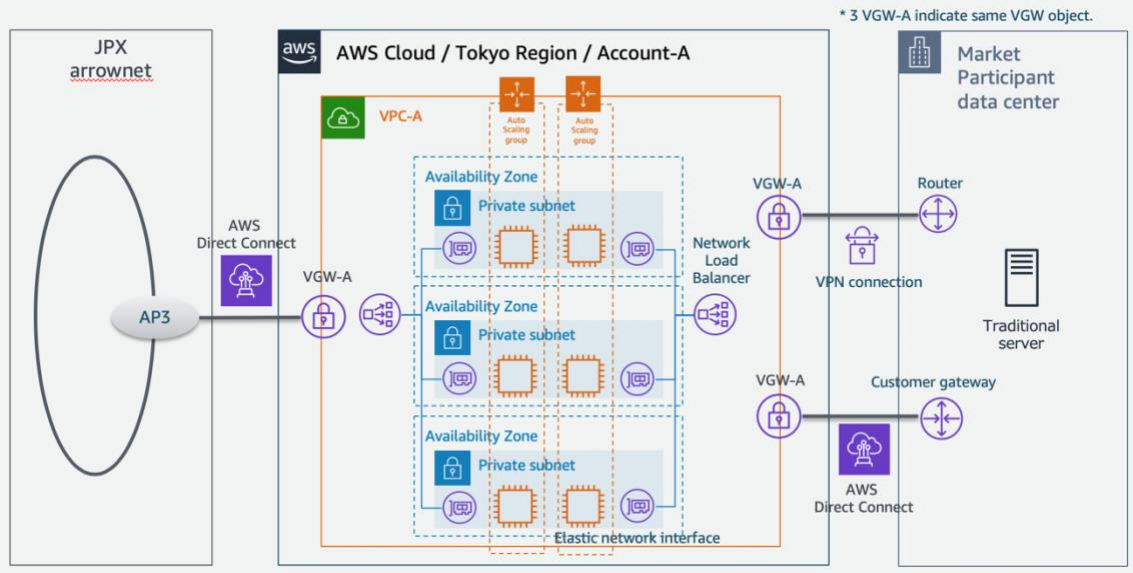


図 4 - トラフィックをオンプレミスのデータセンターへ転送するためのリファレンスアーキテクチャ

設計の原則

シナリオ 1 と同様に、市場参加者には、（JPX とのやり取りに必要がない）内部通信が arrownet に流れることを防止するアーキテクチャであることが求められます。

- JPX トラフィックを処理するには、VPC-A のみ を使用します。このアーキテクチャは、AWS Direct Connect または AWS サイト間 VPN を使用して、VPC-A を介して市場参加者のオンプレミスデータセンターにトラフィックを転送します。
- 市場参加者のインターネットゲートウェイまたは VPN ゲートウェイ は、AWS Direct Connect（arrownet 提供）からプライベート VIF を受け取るため、VPC-A には割り当てません。

アーキテクチャの説明

> **重要：**このシナリオでは、AWS 東京リージョンを使用する必要があります。AWS 東京リージョンの使用は、金融商品取引法で定義されているコンプライアンス要件です。

AWS Transit Gateway を使用すると、VPC-A を構築することなく、AWS Direct Connect または AWS サイト間 VPN 経由でオンプレミスのデータセンターに直接トラフィックを転送できます。この場合、AWS Transit Gateway のルートテーブル設定が正しく設定されていない場合、JPX に許可されていないトラフィックが arrownet 側に流れる可能性があります。この誤ったフローを防ぐには、VPC-A を構築し、VPC-A のプロキシサーバとセキュリティグループの機能を使用してトラフィック制御を実装します。これは、シナリオ *1* で説明されているアーキテクチャと比較して、シナリオ *2* のアーキテクチャの唯一の違いです。

シナリオ 3. ログの集約と分析のアーキテクチャ

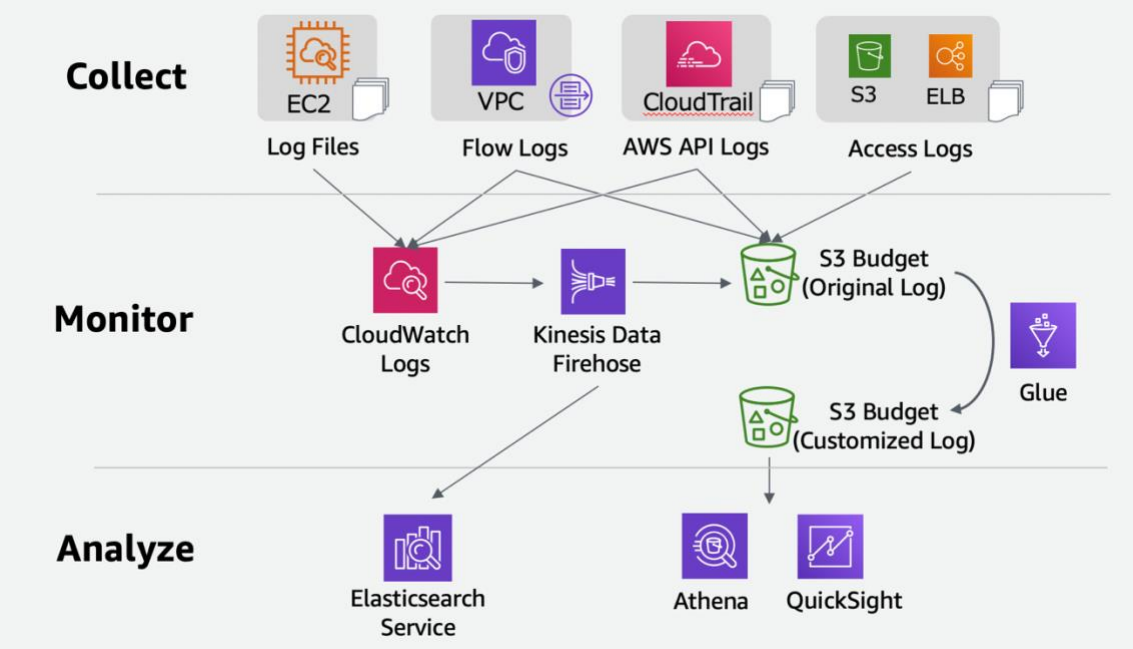


図 5 - ログの集約と分析のアーキテクチャ

設計の原則

内部監査のためにログを集約し、それを分析、そしてインシデントに対応するためのシステムを準備することをお勧めします。上の図 5 は、AWS がログの分析に使用する汎用アーキテクチャを示しています。例えば、ETL は AWS Glue から変える、BI を Amazon Quicksight から変えるなど、市場参加者で利用実績のあるもの、またはセキュリティルールに沿っているものを使うことも可能です。

- シナリオ 1、もしくはシナリオ 2 で必要となるサービスから発生するログや、その操作の結果として記録される AWS Application Programming Interface (API)ログを S3 へ集約するアーキテクチャです。
- S3 に集計されたログを必要なときにいつでも分析できるように、機能を予め準備します。

- ログをカスタマイズするときは、常に元のログのコピーを保持します。

アーキテクチャの説明

シナリオ 1 とシナリオ 2 では、プロキシサーバーログ、VPC ログ、AWS サービスの設定、AWS API 実行ログ（変更が発生したときに生成される）、および NLB/ALB や S3 などのロードバランサーへのアクセス記録は、内部監査の管理やインシデントへの対応に不可欠です。これらのログを S3 バケットに集約するシステムを作成する必要があります。

初期段階 では、AWS モニタリング機能の一部として Amazon CloudWatch Logs によって記録されたログが、Amazon Kinesis Data Firehose を通じてリアルタイムで S3 バケットに集約されます。または、データを約 12 時間間隔でバッチ処理することが許可される場合は、CloudWatch ログ機能を使用してデータを S3 にエクスポートできます。また AWS CloudTrail の実行結果は直接 S3 に保存できます。

AWS Glue は、オリジナルの S3 バケットのログを CSV フォーマット、またはカラムナフォーマットなどデータ処理やデータ分析に適したサイズとファイル形式へ定期的に変換します。次に、このデータを、カスタマイズされたデータ用の別の S3 バケットに保存します。必要なときにデータ分析を実行できるように、Amazon Athena 経由で Amazon QuickSight にデータを事前に準備します。Amazon Athena と Amazon QuickSight を組み合わせて使用すると、これらのオペレーションはサーバーレスで維持されます。また、コストパフォーマンスの点でも優れた選択肢です。これは、Amazon Elasticsearch Service または他のサードパーティーのデータ分析サービスでも変更できます。

AWS Glue などでカスタマイズした場合は、元のログを保持することをお勧めします。これは、データのカスタマイズによるデータの損失を防ぐためです。ログを長期間

保持するには、古いデータを自動的に Amazon S3 Glacier にアーカイブするメカニズムを採用し、長期保存の料金を低くすることができます。

これらの仕組みの実装範囲は、市場参加者毎に判断を委ねます。ただし、最低限、事前に S3 バケットにデータを一括集約することを推奨します。ログが集約されていないことには調査することもできないからです。

AWS Well-Architected フレームワーク

AWS Well-Architected フレームワーク は、クラウドアーキテクトが安全で高パフォーマンス、耐障害性、効率的なインフラストラクチャを構築するのに役立ちます。このフレームワークは 5 つの柱（運用上の優秀性、セキュリティ、信頼性、パフォーマンス効率、コスト最適化）に基づいており、お客様とパートナーがアーキテクチャを評価し、時間とともに拡張できる設計を実装するための一貫したアプローチを提供します。

Well-Architected フレームワーク の 5 つの柱

運用上の優秀性

運用上の優秀性の柱では、ビジネス価値を提供するためのシステムの実行とモニタリング、および継続的にプロセスと手順を改善することに焦点を当てます。主要なトピックには、変更の管理と自動化、イベントへの対応、日常業務をうまく管理するための標準の定義が含まれます。

セキュリティ

セキュリティの柱では、情報とシステムを保護することに焦点を当てます。主要なトピックには、データの機密性と整合性、権限管理における権限の特定と管理、システムの保護、セキュリティイベントを検出する制御の確立が含まれています。

信頼性

信頼性の柱では、ビジネスや顧客の要求に応えるための障害の防止や、障害からの迅速な復旧を行う能力について焦点を当てます。主要なトピックには、設定、プロジェクト間の要件、復旧計画、および変更に対処する方法についての基礎的な要素が含まれます。

パフォーマンス効率

パフォーマンス効率の柱では、IT およびコンピューティングリソースの効率的な使用に焦点を当てます。主要なトピックには、ワークロードの要件に応じた適切なリソースタイプやサイズの選択、パフォーマンスのモニタリング、ビジネスニーズの進展に応じて効率を維持するための情報に基づく決定を行うことが含まれます。

コスト最適化

コスト最適化では、不要なコストの回避に焦点を当てます。主要なトピックには、費用が発生する箇所の把握と管理、最適で正しい数のリソースタイプの選択、時間経過に伴う分析、不要な支出がないビジネスニーズに対応したスケーリングが含まれます。

運用上の優秀性に関する考慮事項

次の表に、Well-Architected フレームワークの運用上の優秀性に関する考慮事項に基づく設計ガイドを示します。

表 2 – 運用上の優秀性に関する考慮事項

考慮事項	よくある質問	ガイド
OPS1	**優先順位はどのように決定すればよいでしょうか?**	• **コンプライアンス要件を評価する** AWS の東京リージョンを使用して、金融商品取引法に定められたコンプライアンス要件に準拠します。 • **脅威の状況を評価する** VPC(Virtual Private Cloud) を分離して、市場参加者の内部通信が JPX に流れないようにします。 • **トレードオフを評価する** 本ガイドでは、最適なコストで安全に使用することを重視しています。最適なバランスを達成するために、設計フェーズ時とテストフェーズ時に、市場参加者のニーズを考慮してください。
OPS2	**ワークロードの状態を理解できるようにするには、ワークロードをどう設計すればよいでしょうか?**	• **ワークロードテレメトリーを実装して設定する** Amazon CloudWatch を使用して NLB や EC2 の稼働ステータスをモニタリングできるように設定します。サードパーティのサービス（Amazon CloudWatch ではそのデータを取得できません）から提供されるサービスとソフトウェアメトリクスも事前に検討してください。

OPS3	**どのようにして欠点を減らし、修正を簡単にし、本番環境へのフローを改善しますか?**	• **変更をテストし、検証する** • **構成管理システムを使用する** • **構築およびデプロイ管理システムを使用する** • **設計標準を共有する** • **コード品質の向上のためにプラクティスを実装する** • **頻繁に、小さく、可逆的な変更を行う** • **統合とデプロイを完全自動化する** 「Infrastructure as Code」という概念を適用し、AWS Cloud Development Kit(CDK) または AWS CloudFormation を使用して、デプロイの自動化をお勧めします。CDK と AWS CloudFormation を使用すると、コードを使用して環境設定に関する情報を定義できます。GitHub または AWS CodeCommit を使用して、設定を管理できます。また、AWS CloudFormation を使用してデプロイを管理することもできます。コードを使用すると、組織は規約とテンプレートを確立し、そのアプローチを標準化し、コードで環境を構築するためのベストプラクティスを共有できます。稼働後の変更は、小さなコードの変更を段階的に適用することで達成できます。これらの変更は、検証と検証のためにテスト環境にデプロイされます。 加えて、人為的なミスは、AWS マネジメントコンソールを使用して手動で環境を作成するときに発生することがあります。コードを使用して環境を構築すると、人為的なミスの余地が減ります。 さらにコードを使用することで、チームの全体的な生産性が向上し、チームメンバー個々のスキルと知識への依存が軽減されます。

考慮事項	よくある質問	ガイド
OPS4	**どのようにデプロイのリスクを軽減しますか?**	• **変更の失敗に備える** • **変更をテストし、検証する** • **デプロイ管理システムを使用する** • **限定的なデプロイを使用してテストする** • **並列環境でデプロイする** • **小規模で可逆的な変更を頻繁にデプロイする** • **統合とデプロイを完全自動化する** • **テストとロールバックを自動化する** 詳細については、[OPS3] を参照してください。
OPS5	**ワークロードをサポートする準備が整っていることはどうすれば確認できるでしょうか?**	• **従業員の対応力の確保** arrownet のテスト系を使用して、開発チームと運用チームが十分なテスト / トレーニングを実施していることを確認します。テストやトレーニングを通じて、以下に示すランブックとプレイブックの有用性を検証することが特に重要です。 • **ランブックを使用して手順を実行する** ランブックとは、具体的な成果を達成するための文書化された手順です。変更を管理し、JPX によるテストイベント、市場参加者によるテストを実行できるように、事前に手順を整備することをお勧めします。 • **プレイブックを使用して問題を特定する** プレイブックは、問題を調査するための文書化されたプロセスです。予想される問題については、事前に手順を整備することをお勧めします。これにより、定期的な AWS メンテナンスに起因する問題、または予期しない arrownet の問題に備えることができます。上記のトレーニングの一環として、プレイブックの有効性を検証することをお勧めします。

OPS6	**ワークロードの正常性をどのように把握しますか?**	• **主要業績評価指標（KPI）を特定する** • **ワークロードのメトリクスを定義する** • **ワークロードメトリクスを収集および分析する** • **ワークロードメトリクスの基準値を設定する** • **ワークロードに対して予想されるアクティビティのパターンを知る** JPX との通信処理を行うアプリケーションのログを収集し、応答レスポンス時間や一定時間あたりの通信量を評価します。このメトリクスから正常性を把握するようにします。また JPX と接続したあとのテスト段階から統計的にこのメトリクスを収集し、通常時の基準値を予め測り、一日の運用をとおしてメトリクスの傾向を事前把握してください。 • **ワークロードの結果にリスクがある場合に警告する** • **ワークロードの異常が検出された場合に警告する** Amazon Simple Notification Service(SNS) を使用して、事前に測定されたメトリクスのベースラインからパフォーマンスが逸脱した場合にアラートを生成します。一部のサードパーティ製ソフトウェア / サービスおよび運用サポートシステムでは、メトリクスとイベントにリンクされたアラートもサポートされています。市場参加者の運用監視の仕組みに応じて、どのように連携できるかを事前に検討してください。 • **KPI とメトリクスの成果の達成度と有効性を検証する** メトリクスの収集と分析、異常時のアラートなど、一連のオペレーションの有効性を定期的に評価します。これらの評価を実施するには、毎月の定例会などを使用できます。

OPS7	**オペレーションの正常性をどのように把握しますか?**	• **主要業績評価指標（KPI）を特定する** • **オペレーションのメトリクスを定義する** • **オペレーションメトリクスを収集し、分析する** • **オペレーションメトリクスの基準値を設定する** • **オペレーションに対して予想されるアクティビティのパターンを知る** テスト時に設定変更の影響をチェックし、デプロイの結果を確認することで、オペレーションが想定どおりに実行されていることを確認します。Amazon CloudWatch を使用してログを確認します。AWS Config を使用して、変更状況を確認します。AWS CloudTrail を使用して、API 実行ログを確認します。現在の結果を過去の結果と比較して、オペレーションが正しく実行されたことを確認できるように、これらの結果を保存することをお勧めします。 • **オペレーションの結果にリスクがある場合に警告する** • **オペレーションの異常が検出された場合に警告する** 不要な操作が実行されるたびにアラートを生成することをお勧めします。AWS Config Rules と AWS CloudTrail には、望ましくない変更が行われたときにアラートを生成する機能があります。たとえば、インターネットゲートウェイが arrownet と通信している VPC にアタッチされた場合や、セキュリティグループの設定に変更が適用された場合などがあります。 • **KPI とメトリクスの成果の達成度と有効性を検証する** トラブル対処後やテストイベント後に、結果の分析を行い、メトリクスに基づいて関係者(ステークホルダー)に修正アクションを提案します。

OPS8	**ワークロードと運用イベントはどのように管理しますか?**	• **イベント、インシデント、問題に対する管理プロセスを使用する** イベント、インシデント、問題に対処するプレイブック（[OPS5] に記載）を事前に準備します。 • **根本原因の分析のプロセスを使用する** 予期しない問題が発生したときに根本原因分析を実行し、対処をプレイブックに追加します。 • **アラートごとのプロセスを使用する** • **ビジネスへの影響に基づき、運用上のイベントを優先します。** • **エスカレーション経路を決定する** • **プッシュ通知を有効にする** 市場参加者のエスカレーションルールに沿い、メンテナンスや問題発生時に自動的に通知する仕組みを事前に検討します。 • **ダッシュボードでステータスを知らせる** イベント中に関係者が現在のステータスを理解できるようにするメカニズムを作成することをお勧めします。AWS では、Amazon CloudWatch ダッシュボードを使用してメトリクスのステータスを追跡します。カスタムメトリクスを提供することもできます。または、Amazon ElasticSearch Service または Amazon QuickSight を使用して、視覚的に状況をダイジェストするダッシュボードを作成します。 • **イベントへの応答を自動化する** 一連の対応プロセスがプレイブックにすでに存在する場合は、イベントへの応答を自動化できる可能性があります。AWS Lambda または AWS Auto Scaling を使用して、アラートにリンクされた自動復旧メカニズムを検討してください。

考慮事項	よくある質問	ガイド
OPS9	**オペレーションを進化させる方法**	• **継続的改善のプロセスを用意する** • **フィードバックループを実装する** • **改善の推進要因を定義する** • **洞察を検証する** • **オペレーションメトリクスのレビューを実行する** • **教訓を文書化して共有する** • **改善を行うための時間を割り当てる** [OPS1-8] で説明されている内容に沿って、定期的な運用改善ができるように組織的な仕組みを構築することをお勧めします。

セキュリティに関する考慮事項

次の表に、Well-Architected フレームワークのセキュリティに関する考慮事項に基づく設計ガイドを示します。

表 3 - セキュリティに関する考慮事項

考慮事項	よくある質問	ガイド
SEC1	**認証情報と認証をどのように管理していますか?**	• **アイデンティティおよびアクセス管理要件を定義する** 市場参加者の組織が定義するセキュリティルールに従い、ID および権限管理の要件の整理を実施します。 • **AWS ルートユーザーを保護する** AWS ルートユーザーの権限は強力であるために、このユーザーを保護するために多要素認証(Multi-Factor Authentication：MFA)を設定し、必要最小限の利用に限定することを推奨します。 • **MFA の使用を義務化する** arrownet を使用して AWS リソース処理通信を操作する権限を持つ AWS アカウントの IAM ユーザーに MFA を実装することをお勧めします。これは、arrownet 通信設定が他の市場参加者に影響を与える可能性があるためです。これにより、重要な AWS リソースを誤って変更または削除することを防止できます。 • **アクセスコントロールの義務化を自動化する** • **一元化されたフェデレーションプロバイダーと統合する** ID プロバイダーとディレクトリーサービスを統合し、どのユーザーがアクセスをしたかを明確にすることを推奨します。 • **パスワード要件を義務化する** 強力なパスワードポリシーを実装することをお勧めします。 • **認証情報を定期的にローテーションする** ログインパスワードの有効期限を設定することをお勧めします。これにより、許可されていないユーザーが不要なアクセスから保護されます。 • **認証情報を定期的に監査する** AWS Identity and Access Management から IAM ユーザーの認証情報レポート(Credential Report)を取得し、定期的な監査を実行することをお勧めします。

考慮事項	よくある質問	ガイド
SEC2	**人為的なアクセスをどのように制御していますか?**	• **人為的なアクセス要件を定義する** [SEC1] と同様に、市場参加者の組織によって定義されたセキュリティルールに従い、職務機能に基づいて AWS IAM ロールのアクセス要件を設定することをお勧めします。構築と運用に関わるユーザーに加えて、セキュリティインシデントへの対応に関わるユーザーや監査に関わるユーザーに対して、適切な IAM ロールを事前に準備します。 • **最小限の権限を付与する** 許可されていないアクセスのリスクを軽減するために、定義した最小限の特権のみをユーザーに付与します。 • **各個人に一意の認証情報を割り当てる** • **ユーザーライフサイクルに基づいて認証情報を管理する** • **認証情報管理を自動化する** • **ロールまたはフェデレーションを通じてアクセス権を付与する** これを実現するには、一元化されたフェデレーションプロバイダー（[SEC1] に記載）と統合することをお勧めします。

考慮事項	よくある質問	ガイド
SEC3	**プログラムによるアクセスをどのように制御していますか?**	• **プログラムによるアクセス要件を定義する** • **最小限の権限を付与する** • **認証情報管理を自動化する** • **ロールまたはフェデレーションを通じてアクセス権を付与する** • **動的認証を実装する** プログラムによるアクセスを通じて AWS 環境を運用している場合でも、[SEC1-2] で説明されている内容を実装することをお勧めします。 • **各コンポーネントに一意の認証情報を割り当てる** 安全なプログラムによるアクセスを許可するには、IAM ユーザーまたは静的アクセスキーの代わりに IAM ロールまたはフェデレーションを使用することをお勧めします。

SEC4	**セキュリティイベントをどのように検出し、調査していますか?**	• **ログの要件を定義する** 市場参加者の組織によって定義されたセキュリティルールを満たすログの保管とアクセスコントロール要件を確立します。 • **メトリクスの要件を定義する** AWS CloudTrail を使用して AWS の操作履歴を収集することをお勧めします。Amazon CloudWatch Logs でログメトリクスを評価することをお勧めします。ログメトリクスの例として、特定のログが短期間に頻繁に生成されていないかどうかが挙げられます。また、また GuardDuty を設定することを推奨し、不正な API の呼び出しや予期しないリソースのアクセスに対するモニタリングをお勧めします。 • **アラートの要件を定義する** • **主要な指標に関するアラートを自動化する** セキュリティインシデントに関するアラートを受け取る部門と、コミュニケーションの方法を定義します。CloudWatch Logs でメトリクスの異常が検出された場合や、Amazon GuardDuty で許可されていないアクティビティが識別された場合に、これらの定義に基づいてアラートを送信するには、Amazon SNS を使用することをお勧めします。 • **サービスとアプリケーションのログ記録を設定する** Amazon CloudWatch ログ、VPC フローログ、AWS CloudTrail を使用してログを収集します。 • **ログを一元的に分析する** これらのログは S3 で一元化、集計、分析することをお勧めします。また、AWS Security Hub を使用して可視性を向上させることをお勧めします。 • **調査プロセスを開発する** [OPS5] および [OPS7] の内容を参照してください。

考慮事項	よくある質問	ガイド
SEC5	**新しいセキュリティ脅威に対してとどのように防御していますか?**	• **組織要件、法的要件、コンプライアンス要件に関する最新情報を入手する** • **セキュリティ脅威に関する最新情報を入手する** • **脅威モデルを使用してリスクを定義し優先順位付けする** 市場参加者の組織が定義するセキュリティルールに従い、実装すべき要件を確立します。 • **セキュリティのベストプラクティスに関する最新情報を入手する** • **新しいセキュリティサービスとセキュリティ機能を定期的に評価する** • **新しいセキュリティサービスとセキュリティ機能を実装する** [PERF6] を参照し、Amazon Inspector を使用して不明なアクセスを検出することを検討してください。
SEC6	**ネットワークをどのように保護していますか?**	arrownet が閉域ネットワークサービス であると仮定すると、このリファレンスアーキテクチャは市場参加者のワークロードの非武装地帯 (DMZ) として機能し、市場参加者のネットワークを保護します。また、セキュリティグループまたはネットワークアクセスコントロールリスト (NACL) を使用して、不要な通信を防止することで、市場参加者のワークロードを保護することもできます。また、ネットワーク保護の観点からは、[SEC1] および [OPS3] を参照し、人為的ミス（AWS マネジメントコンソールを使用した手動操作の制限など）を防ぐための対策を検討してください。

考慮事項	よくある質問	ガイド
SEC7	**コンピューティングリソースをどのように保護していますか?**	arrownet は閉域ネットワークサービスですが、arrownet または市場参加者からの不正なアクセスを防ぐために EC2 の OS にパッチを定期的に適用します。AWS Systems Manager が利用可能であれば、セキュリティパッチの適用を自動化することを検討してください。
SEC8	**データをどのように分類していますか?**	このドキュメントにあるアーキテクチャは、JPX との市場参加者の取引に関するデータは蓄積しません。蓄積すべきログを事前に検討してください。なお、これらのログは JPX サービスとの通信から生成されるため、サービスタイプによっては、市場参加者の視点からの機密情報が含まれる場合がありますので留意ください。
SEC9	**保管中のデータをどのように保護していますか?**	ログ保管のために暗号化することをお勧めします。AWS Key Management Service のカスタマー管理型キーを使用して、ログにアクセスできるサービスと IAM アカウントを制限することを検討してください。これにより、誤ってログ情報を変更、削除、移動する可能性が制限されます。さらに強力な保護のために、ログが保存されるストレージサービス（S3 など）への直接アクセスを防止する設定も適用できます。
SEC10	**伝送中のデータをどのように保護していますか?**	JPX が提供するサービスの定義された通信要件に従います。

SEC 11	**セキュリティインシデントにどのように対応していますか?**	• **重要な人員と外部リソースを特定する** • **ツールを特定する** • **インシデント対応計画を策定する** • **ツールを事前デプロイする** 詳細については、[OPS5] を参照してください 。 • **封じ込め機能を自動化する** 市場参加者側の異常が arrownet へ流れないように封じ込める施策を事前に検討ください。例えば、AWS の AutoScaling の機能を停止し、プロキシサーバを構成する EC2 を停止する手段などが挙げられます。 • **フォレンジック機能を確認する** AWS には VPC トラフィックミラーリング機能があり、EC2（プロキシサーバーのセットアップに使用）インスタンスのネットワークトラフィックをキャプチャおよびミラーリングできます。また、市場参加者と arrownet 間の定期的なトラフィック、市場参加者のセキュリティルール、およびコスト効率を考慮して、実装方法と場所の両方を考慮してください。 • **アクセスを事前準備する** [SEC2] の説明を参照してください。 • **ゲームデーを実施する** 定期的な訓練を行うことで、まだ組織異動の結果が更新されていないエスカレーションパス（組織内の連絡など）を明らかにしたり、特定の対処方法がないことを明らかにしたりできます。[OPS5]、[OPS7]、[OPS8]、[OPS9] の説明を参照しながら、訓練の実施を検討してください。

信頼性に関する考慮事項

次の表に、Well-Architected フレームワークの信頼性に関する考慮事項に基づく設計ガイドを示します。

表 *4* - 信頼性に関する考慮事項

考慮事項	よくある質問	ガイド
REL1	**サービス制限をどのように管理していますか？**	AWS アカウント毎にプロビジョニングできるリソースには、クォータ（以前は制限と呼ばれていました）があります。AWS Service Quotas でプロビジョニングできるリソースを確認できます。arrownet とのトラフィック量が大きい場合は、大きな EC2 インスタンスを使用する可能性があります。その場合は、クォータに十分に注意を払ってください。また、AZ の障害を含むサービスの制限やフェイルオーバーに対応できるかどうかを事前に確認してください。足りなくなる可能性がある場合は、事前にクォータを引き上げてください。
REL2	**ネットワークトポロジをどのように管理していますか?**	本リファレンスアーキテクチャガイドのシナリオ別のアーキテクチャを参考に、arrownet と市場参加者の AWS 環境との接続、さらにオンプレミス環境との接続において、可用性を高める手段を検討ください。

考慮事項	よくある質問	ガイド
REL3	**システムが需要の変化にどのように対応していますか?**	arrownet 自体に通信量のスロットリング機能がありますが、[OPS6] の内容に従って、常にメトリクスをモニタリング、累積、分析し、トラフィックに応じて EC2 インスタンスの数とサイズを最適化し続けることをお勧めします。Auto Scaling 機能は、インスタンスの置き換えや、複数のアベイラビリティーゾーンにわたるベースキャパシティーの確保に役立ちます。 接続開始時や、変更時は、JPX arrownet version2.0 のガイドラインに従い、負荷試験の実行をお勧めします。
REL4	**リソースをどのようにモニタリングしていますか?**	[OPS6] の内容に従って、Amazon CloudWatch メトリクスとログを使用して arrownet との通信ステータスをモニタリングします。AWS Service Health Dashboard と Personal Health Dashboard を使用して、AWS サービス自体でサービス状況もモニタリングします。
REL5	**変更をどのように実施していますか?**	[OPS3] の内容 に従って、AWS CDK/AWS CloudFormation によるコードへ変更を反映し、これらの変更のデプロイを自動化します。

考慮事項	よくある質問	ガイド
REL6	**データをどのようにバックアップするか?**	• **バックアップする必要があるデータをすべて特定し、バックアップやソースからのデータの複製を実行する** 本ガイドで記載しているアーキテクチャの範囲では、EC2 で利用している EBS のバックアップには、EBS のスナップショット機能などが利用可能です。 • **データのバックアップまたはソースからのデータの複製を自動的に実行する** AWS Backup を使用して、複数世代のスナップショットが残るように、設定を作成します。すでにジョブ管理機能がある場合は、ジョブから AWS コマンドラインインターフェイス (CLI) を使用し、スナップショットとその生成を定期的に管理するバッチアプリケーションを実装します。 • **データの定期的な復旧を行ってバックアップの完全性とプロセスを確認する** これは、[OPS5] のプレイブックの準備とそのユーティリティチェックに関連しています。EC2 が取得されたスナップショットから復元できることを事前に確認することをお勧めします。 • **バックアップを保護または暗号化するか、データが安全なソースから複製できることを確認する** EBS が暗号化されているためにスナップショットも、暗号化されます。

考慮事項	よくある質問	ガイド
REL7	**どのようにしてシステムがコンポーネントのエラーに耐えるか?**	• **ワークロードのすべてのレイヤーをモニタリングしてエラーを検知する** 外形監視として、 arrownet を介してアクセス可能な JPX の各システム / サービスによって提供されるヘルスチェック機能を使用して、システム / サービスが市場参加者の視点から正常に機能しているかどうかを監視します。 • **可用性に影響を及ぼすイベントが発生したら通知を送信する** [OPS6] と [OPS7] の説明に従って、ワークロードと操作の正規性がモニタリングされ、エラーが発生した場合に通知が送信されます。 • **疎結合の依存関係を実装する** • **該当するハードな依存関係をソフトな依存関係に変換するため、グレイスフルデグラデーションを実装する** • **単一の場所を求める技術的な制約がワークロードの一部または全体に存在するため、復旧全体を自動化する** • **複数の場所にワークロードをデプロイする** • **すべてのレイヤーの修復を自動化する** 市場参加者がこの接続を重要なワークロードとして使用する場合、JPX より 2 つ以上の VIF の提供を受け、AWS Direct Connect ルートの異常に備えます。また、本ガイドのリファレンスアーキテクチャに示すように、複数のアベイラビリティーゾーン間で自動的に復旧するプロキシサーバーを設定します。

考慮事項	よくある質問	ガイド
REL8	**弾力性をどのようにテストしていますか?**	ガイドは弾力性を提供するように設計 されていますが、[OPS5] および [OPS8] で説明されているように、十分なテストを事前に実行することをお勧めします。
REL9	**災害対策をどのように計画していますか?**	Well-Architected フレームワークにあるこの質問に対する答えは、本ガイドの執筆時点 では、arrownet は関東地域以外の広域災害対策をサポートしていないため、「ワークロードには適用されません」となります。

パフォーマンス効率に関する考慮事項

次の表に、Well-Architected フレームワークのパフォーマンス効率に関する考慮事項に基づく設計ガイドを示します。

表 5 - パフォーマンス効率に関する考慮事項

考慮事項	よくある質問	ガイド
PERF1	**最も良いパフォーマンスのアーキテクチャをどのように選択していますか?**	この実装ガイドに従って設計および構築 することをお勧めします。また、負荷テストを実施するには、JPX によるガイドに従う必要があります。
PERF2	**コンピューティングソリューションをどのように選択していますか?**	通信パフォーマンスを評価しながら、市場参加者の目的に応じた EC2 インスタンスサイズ を選択します。設定オプションとメトリクスの取得の詳細については、「アーキテクチャの説明」を参照してください。さらに、[OPS6] および [OPS7] で説明されているように、メトリクスに基づいて定期的な評価を実行し、最適なインスタンスサイズを選択することをお勧めします。

考慮事項	よくある質問	ガイド
PERF3	**ストレージソリューションをどのように選択していますか?**	プロキシサーバーとして設定する EC2 インスタンスで使用される EBS パフォーマンスを 評価し、市場参加者の目的に基づいて EBS ボリュームタイプを選択します。EBS 暗号化設定オプションの詳細については、[SEC8-9] を参照してください。最適な EBS のボリュームタイプに合わせて調整するインスタンスサイズを考慮しながら、メトリクスに基づいて定期的な評価を実行することをお勧めします。
PERF4	**データベースソリューションをどのように選択していますか?**	この実装ガイドは、データベースソリューションを利用しません。したがって、Well-Architected フレームワークに関するこの質問に対する答えは、「ワークロードには適用されません」となります。

考慮事項	よくある質問	ガイド
PERF5	**ネットワークソリューションをどのように選択していますか?**	• **ネットワーキングがパフォーマンスに与える影響を理解する** • **使用可能な製品オプションを理解する** • **使用可能なネットワーク機能を評価する** • **暗号化オフロードと負荷分散を活用する** この実装ガイドに従って設計および構築することをお勧めします。 • **最小限の Network ACL を使用する** • **パフォーマンスを高めるネットワークプロトコルを選択する** プロキシサーバーにアタッチされたセキュリティグループを使用し、NACL をできる限り使用しないようにすることをお勧めします。各 JPX のサービスが提供する通信要件に従って、セキュリティグループの通信許可を設定ください。 • **ネットワーク要件に基づいてロケーションを選択する** 金融商品取引法を遵守する必要があります。「アーキテクチャの説明」を参照し、適切なリージョンを選択します。 • **メトリクスに基づいてネットワーク設定を最適化する** EC2 インスタンスサイズのチューニングとなるため、[PERF2] の説明を参照してください。

考慮事項	よくある質問	ガイド
PERF6	**ワークロードを進化させるためにどのように新機能を取り込んでいますか?**	当社は、サービスにおいて継続的なイノベーションを実施しています。詳細については、 *AWS* の最新 情報と *AWS* ブログ を定期的に参照してください。AWS ソリューションアーキテクトに相談することもできます。新しい機能リリースが既存のワークロードに与える影響を評価し、アクティブなアプローチを取り、ワークロードパフォーマンスを進化させるプロセスを実装することをお勧めします。
PERF7	**リソースが正常に動作していることを確認するためにどのようにモニタリングしていますか?**	[OPS3] から [OPS9] の記載 を参照してください。
PERF8	**パフォーマンスを向上させるために、トレードオフをどのように利用していますか?**	本ガイド では、EC2 インスタンスサイズのチューニングになります。通信性能が高いインスタンスを使えば、コストも増加する可能性があります。詳細については、[COST4] を参照してください。

コスト最適化に関する考慮事項

次の表に、Well-Architected フレームワークのコスト最適化に関する考慮事項に基づく設計ガイドを示します。

表 6 - コスト最適化に関する考慮事項

考慮事項	よくある質問	ガイド
COST1	**使用状況をどのように管理しますか?**	• **組織の要件に基づいてポリシーを策定する** • **アカウント構造を実装する** • **コストコントロールを実装する** • **プロジェクトのライフサイクルを追跡する** この実装ガイド では、専用の AWS アカウントを使用して、arrownet トラフィックを処理します。この AWS アカウントで発生したすべてのコストは、arrownet との通信に関連しているため、コスト管理が容易になります。 • **グループとロール を実装する** [SEC1-3] の内容を参照してください。
COST2	**使用状況とコストをどのようにモニタリングしますか?**	[COST1] で説明されている ように、この AWS アカウントで発生したコストはすべて arrownet との通信に関連しています。AWS Cost Explorer でコストを追跡し、AWS Budgets でコスト管理を計画することをお勧めします。コストが急に変化すると、環境に異常が生じるトリガーとして見なすことができます。AWS Budgets をトリガーに、SNS から管理者に通知を送信します。

考慮事項	よくある質問	ガイド
COST3	**不要なリソースをどのように削除しますか?**	[OPS3-8] で説明されている ように、Runbook/playbook の手順に従って不要なリソースを削除することをお勧めします。
COST4	**サービスを選択するとき、どのようにコストを評価しますか?**	プロキシサーバーを最適化し、[PERF2]、[PERF3]、[PERF5] で説明されているようにコスト分析を実施することをお勧めします。
COST5	**リソースタイプとサイズを選択する際、どうすればコスト目標を達成できるでしょうか?**	[COST4] の内容 を参照してください。

考慮事項	よくある質問	ガイド
COST6	**コストを削減するには、料金モデルをどのように使用したらよいでしょうか?**	• **料金モデルの分析を実行する** • **低カバレッジでさまざまな料金モデルを実装する** • **このワークロードのすべてのコンポーネントに対して料金モデルを実装します。** インスタンスのキャパシティーを予約し、コストを最適化することをお勧めします 。これら両方を同時に達成するために、本ガイドでは、リザーブドインスタンスの使用をお勧めします。 • **コストに基づいてリージョンを選択する** コンプライアンスを維持するには、東京リージョン を選択する必要があります。
COST7	**データ転送料金についてどのように計画していますか?**	[COST4] の内容を参照してください。
COST8	**リソースの供給と顧客の需要をどのように一致させていますか?**	本ガイド では、Feasibility Study と設計計画によってインスタンスタイプが事前に選択されており、その結果、日常の使用ではリソースの需要が急速に変動するとは思われません。ただし、中長期にわたって、arrownet でトラフィックメトリクスを評価する必要があります。結果に応じて、インスタンスタイプを変更してリソースの供給と実際のトラフィック需要のバランスを取るなどのアクションを実行します。

考慮事項	よくある質問	ガイド
COST9	**新しいサービスをどのように評価していますか?**	[PERF6] の内容 を参照してください。

まとめ

AWS を使用して arrownet に接続すると、ビジネスの俊敏性の向上、人的リソースの最適化、投資の最大化に役立ちます。

ビジネスの俊敏性

インフラストラクチャとそのリソースはいつでも割り当ておよび変更できるため、市場の変動に対する耐性を高めることができます。これにより、利用者が市場にサービスをもたらすまでの時間も短縮されます。

人的リソースの最適化

AWS クラウドを使用することで、エンジニアはさまざまなタスク（データセンターへの入館申請作業など）から解放され、本来やるべきことに集中できます。さらに、物理的な回線申請はなくなり、利用者は関係する部門 / 会社とのやり取りから解放されます。

投資の最大化

市場の変動を先読みし、通常必要とする IT リソースの数倍を確保するような IT 投資を排除することにより、利用者の投資計画が最適化されます。利用者は、前払い料金なしで、必要なだけの IT リソースを使用できます。

この実装ガイドでは、arrownet のような市場インフラストラクチャをユーザーが安全に使用するために必要な事前の準備と対策について取り上げます。本ガイドでは、AWS クラウドの利点を活用しながら、arrownet との安全な接続を確立できます。

寄稿者

本ドキュメントの寄稿者は以下のとおりです。

- 澤野佳伸、ソリューションアーキテクト、アマゾン ウェブ サービス

コメントとフィードバック

本ガイドに関するフィードバックがある場合は、以下の E メールアドレスを使用してお問い合わせください。

E メール：aws-jp-refarchguide-jpxarrownet@amazon.com

ドキュメントの改訂

日付	説明
2020 年 5 月	初版発行

その他のリソース

詳細については、以下のソースを参照してください。

- ***AWS* による *Well-Architected* のフレームワーク**
  https://aws.amazon.com/jp/architecture/well-architected/
- **オンラインセミナーのドキュメント（日本語版）**
  https://aws.amazon.com/jp/aws-jp-introduction/aws-jp-webinar-service-cut/
- **Japan Exchange Group, Inc(JPX)**
  https://www.jpx.co.jp/
- **arrownet**
  https://www.jpx.co.jp/systems/network/index.html

付録

AWS を使用し、arrownet 接続サービスを構築するには、AWS サービスコンポーネントについて理解する必要があります。この付録では、本ガイドで使用されているサービスに関するリファレンス情報を提供します。

1. Amazon Athena- https://aws.amazon.com/athena/
2. Amazon CloudWatch- https://aws.amazon.com/cloudwatch/

3. Amazon Elastic Compute Cloud(EC2)- https://aws.amazon.com/ec2/
4. Elastic Load Balancing- https://aws.amazon.com/elasticloadbalancing/
5. Amazon Elasticsearch Service- https://aws.amazon.com/elasticsearch-service/
6. Amazon GuardDuty- https://aws.amazon.com/guardduty/
7. Amazon Inspector- https://aws.amazon.com/inspector/
8. Amazon Kinesis- https://aws.amazon.com/kinesis/
9. Amazon QuickSight- https://aws.amazon.com/quicksight/
10. Amazon Simple Notification Service(SNS)- https://aws.amazon.com/sns/
11. Amazon Simple Storage Service(S3)- https://aws.amazon.com/s3/
12. Amazon S3 Glacier- https://aws.amazon.com/glacier/
13. Amazon Virtual Private Cloud(VPC)- https://aws.amazon.com/jp/vpc/
14. AWS Auto Scaling- https://aws.amazon.com/jp/autoscaling/
15. AWS Backup- https://aws.amazon.com/backup/
16. AWS Budgets- https://aws.amazon.com/aws-cost-management/aws-budgets/
17. Amazon AWS Cloud Development Kit(CDK)- https://docs.aws.amazon.com/cdk/latest/guide/what-is.html
18. AWS CloudTrail- https://aws.amazon.com/cloudtrail/
19. AWS CloudFormation- https://aws.amazon.com/cloudformation/
20. AWS CodeCommit https://aws.amazon.com/codecommit/
21. AWS コマンドラインインターフェイス (CLI)- https://aws.amazon.com/cli/

22. AWS Config- https://aws.amazon.com/config/
23. AWS Cost Explorer- https://aws.amazon.com/aws-cost-management/aws-cost-explorer/
24. AWS Direct Connect- https://aws.amazon.com/jp/directconnect/
25. AWS Glue- https://aws.amazon.com/glue/
26. AWS Identity and Access Management(IAM)- https://aws.amazon.com/iam/
27. AWS Key Management Service(KMS)- https://aws.amazon.com/kms/
28. AWS Lambda- https://aws.amazon.com/lambda/
29. AWS Security Hub- https://aws.amazon.com/security-hub/
30. AWS Systems Manager- https://aws.amazon.com/systems-manager/
31. AWS Transit Gateway- https://aws.amazon.com/jp/transit-gateway/