opentext™ | Cybersecurity

ポジションペーパー

NetIQ Access Manager のクラウドインフラストラクチャへの移行

目次

スムーズな移行のために

本資料では、NetIQ Access Manager をクラウドベースの導入モデルに移行する際のプロセスと考慮すべき要素について説明します。

NetIQ Access Manager のアーキテクチャ

従来のオンプレミスへの NetIQ Access Manager by OpenText™ の導入とクラウドベースの導入には多くの違いがあります。この 2 つを比較する前に、NetIQ Access Manager の機能コンポーネントについて理解しておく必要があります。NetIQ Access Manager には、管理コンソール、Analytics Server、Identity Server、Access Gateway という 4 つの高レベル機能コンポーネントがあります。

管理コンソールは、システムの設定の管理、保存、配布に使用します。複数の管理コンソールでデータストアの複製を共有することで、システムが本質的にフォールトトレラントになり、単一障害点を排除できます。

Analytics Server は、監査データと運用データをキャプチャします。設定可能なダッシュボードと強力なレポート機能を備えています。Analytics Server によって、アクセス管理システムのセキュリティ、使用率、パフォーマンスの可視性が得られます。

Identity Server は、NetIQ Access Manager に統合された Web ワークロードへのアクセスを認証します。フェデレーションプロトコルを使用してユーザーを認証するため、一般にアイデンティティプロバイダー (IDP) とも呼ばれます。すべての一般的なフェデレーションプロトコルをサポートし、フェデレーション接続を仲介できます。認証フレームワークはプラグインアーキテクチャを使用しています。そのため、既製のオプションのいずれかでニーズが満たされない場合は、カスタムの認証方法を簡単に追加できます。Identity Server には管理コンソールが必要ですが、Access Gateway は必要ありません。

Access Gateway サービスはリバースプロキシであり、その主な目的は、Identity Server を直接統合できないレガシーアプリケーションと統合することです。アプリケーションまたはアプリケーションプラットフォームで SAML2、OAuth、WS-Federation などのプロトコルを使用できる場合、Access Gateway はオプションになります。ただし、これを使用すべき理由がいくつかあります。Access Gateway を使用すれば、セキュリティが一段と強化されます。アプリケーションサーバーよりも攻撃対象領域が小さく、潜在的な脆弱性が低くなることが期待できます。アプリケーションの依存関係を考慮する必要がないため、パッチ適用と脆弱性の軽減対応を簡略化できます。リバースプロキシで Web コンテンツをキャッシュできるため、システムパフォーマンスが向上します。また、要求されたパスでトラフィックをルーティングできるため、複数のバックエンドアプリケーションとサービスを単一の統合アプリケーションとして提供できます。最後に、Access Gateway では、ポリシーベースの集中型アクセス制御が提供され、これによってアプリケーションが備えるセキュリティを置き換えたりレベルを強化したりすることができます。

別の製品ですが、NetIQ Advanced Authentication by OpenText は、NetIQ Access Manager と合わせて導入し、包括的な多要素認証機能を利用する場合がよくあります。Advanced Authentication は、オンプレミスまたはクラウドベースの NetIQ Access Manager 導入環境とシームレスに統合できます。

NetIQ Access Manager バージョン 4.x では、2 つの導入モデルがサポートされています。1 つ目は、製品を構成するすべてのサービスが 1 つのアプライアンスノードにまとめて導入されるソフトアプライアンスモデルです。2 つ目のモデルは、Linux または Windows ベースのオペレーティングシステムを実行している別々のノードにサービスを個別に導入するものです。NetIQ Access Manager バージョン 5.0 では、新しい Docker コンテナベースの導入モデルを提供しており、これはクラウド導入に最適で、アプライアンス導入モデルに代わるものです。コンテナベースのモデルは、アプライアンスモデルよりもさらに柔軟です。必要に応じて、必要なコンテナを 1 つのオペレーティングシステムノードにまとめて導入できます。これにより、以前のアプライアンスモデルで課されていた制限なしに、コンテナとノードの数を動的にスケーリングできます。本書ではこれ以降、非コンテナモデルを「レガシー」モデルと呼びます。レガシーモデルによる一般的な導入を次の図に示します。

NetIQ Advanced Authentication は、オンプレミスまたはクラウドベースの NetIQ Access Manager 導入環境とシームレスに統合できます。

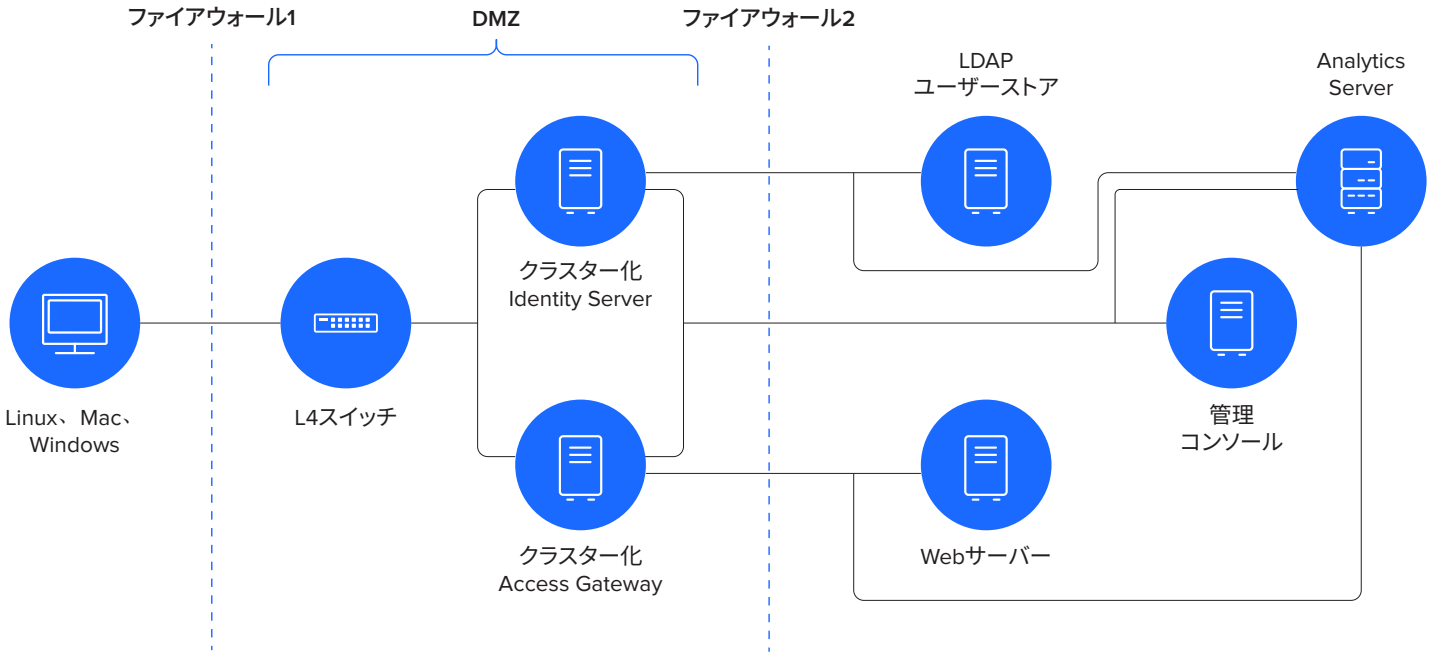

図 1. NetIQ Access Manager のレガシーモデル

このモデルでは、NetIQ Access Manager のコアサービスの冗長インスタンスを実行する複数のサーバー (物理サーバーまたは仮想サーバー) が存在します。サービスはいずれもノードごとに 1 つのインスタンスだけを実行できます。管理コンソールサービスと Identity Server サービスを同じノード上で実行することは可能ですが、セキュリティ上の懸念からこの構成は本番環境ではあまり使用されません。

クラウドインフラストラクチャ上で実行されている仮想マシン (インスタンス) でレガシーモデルを使用することも可能です。オンプレミスの導入とは少し異なりますが、大局的には同じアーキテクチャです。この方法で導入すると、インフラストラクチャにある程度の自動スケーリングを追加することもできます。このオプションは、コンテナへの移行準備が整っていない組織にとっては魅力的ですが、クラウドベースの導入に最適なオプションではありません。柔軟性を大きく諦めることになります。また、仮想マシンインスタンスに十分なリソースを割り当てる必要があるため、このモデルではコストが高くなりがちです。

クラウド導入に最適なオプションは、Docker コンテナと Kubernetes オーケストレーションに基づくクラウドネイティブアーキテクチャを使用することです。

クラウド導入のためにより適したオプションは、Docker コンテナ、Kubernetes オーケストレーション、Helm 構成管理に基づく最新のクラウドネイティブアーキテクチャを使用するものです。これらの技術には多くの利点がありますが、クラウドの経験が乏しい組織にとっては、その活用は大きな変化かもしれません。しかし、これらは主要なクラウドネイティブサービスすべての背後にあるテクノロジーであり、エンタープライズレベルで採用が急激に進んでいます。組織にサポートする準備が整っていないという確信があるということでない限り、この導入モデルを追求するのが妥当です。

これらのテクノロジーを詳しく見て、そのメリットを確認しましょう。

NetIQ Access Manager でのコンテナテクノロジーの使用の概要

Docker コンテナ

コンテナとは何か？ コンテナは、アプリケーションとそのすべての依存関係を含めてカプセル化したランタイム環境です。コンテナを使用すると、オペレーティングシステムやインフラストラクチャに依存することなくアプリケーションを管理できます。これは非常に強力な概念です。基盤となるインフラストラクチャからアプリケーションのサポートを分離することで、両者をより効果的に管理できます。ソフトウェアのリリースとアップグレードのサイクルを短縮でき、システムリソースをより効率的に管理できるほか、アプリケーションの移植性も大幅に向上します。何よりも、OpenText™ のエンジニアが、NetIQ Access Manager コンテナ内のすべての要素を完全に制御およびサポートします。アプリケーションをアップグレードするには、コンテナの新しいバージョンを導入するだけですみます。

Docker は、コンテナの作成と使用に必要なツールとフレームワークを提供するオープンソースプロジェクトです。事実上の業界標準で、すべての主要なクラウドベンダーでサポートされています。Docker を使用すると、開発者のワークステーション上でもパブリッククラウドプラットフォームでも、全く同じアプリケーション構成を実行できます。実際、2 つの環境間でのアプリケーションの移動は数分でできます。

次の図は、従来の仮想化とコンテナのアーキテクチャの違いを示しています。コンテナの方が、小さく、移植性が高く、必要なリソースも少なくて済みます。ホストシステムとゲスト OS の両面で、特定のオペレーティングシステムへの依存を排除できます。コンテナにはオペレーティングシステムの全体または一部が含まれている場合もありますが、コンテナ開発者がそのコードを管理します。運用担当者がそのオペレーティングシステムを保守する必要はありません。完全に透過的になります。

Kubernetes は、多数のアプリケーション、サービス、ホストノードで構成されるシステム全体を管理するためのフレームワークを提供します。

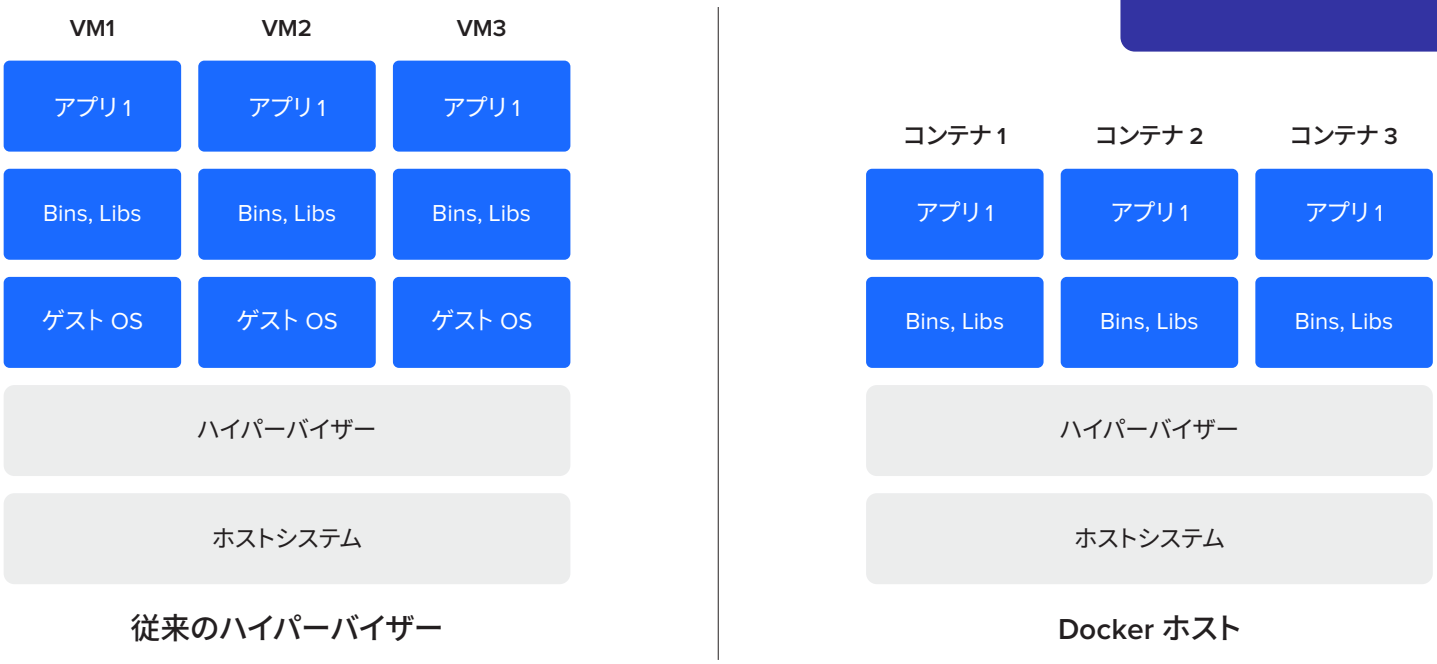

図 2. 従来の仮想化とコンテナの比較

Docker を使用すると、アプリケーションの導入と更新が非常に簡単になります。1 つのコマンドでアプリケーション全体を導入できます。1 つのコマンドでアプリケーションを最新バージョンに更新することもできます。その更新に問題があれば、構成を以前のバージョンにロールバックすることも 1 つのコマンドで行うことができます。

Kubernetes によるコンテナオーケストレーション

コンテナベースの NetIQ Access Manager の導入は、別々のコンテナに格納された複数のサービスで構成されます。システムを拡張すると、コンテナの数が増加します。コンテナ自体の管理に加えて、ネットワーク、ロードバランシング、スケーリングなどのシステムの側面を管理する必要があります。ここで Kubernetes が役に立ちます。Docker が、コンテナベースのアプリケーションをホストノード上で効果的に管理するためのツールを提供するのに対し、Kubernetes は、多数のアプリケーション、サービス、ホストノードで構成されるシステム全体を管理するためのフレームワークを提供します。

Kubernetes は、コンテナの論理的なグループ化を「Pod」と呼ばれるセットに編成します。各 Pod は、リソースの可用性に基づいてホストノードに導入できるユニットとして管理されます。Kubernetes は Pod のステータスを監視し、必要に応じて Pod の再起動や移動ができます。

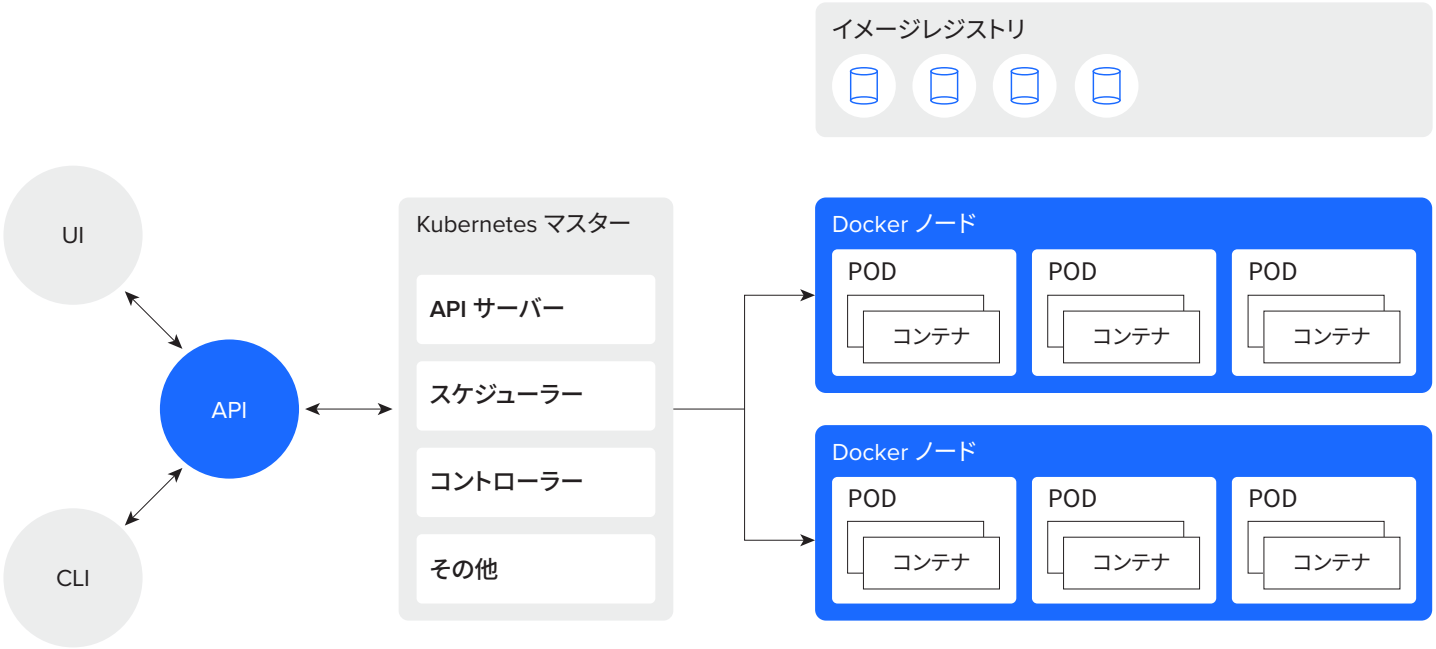

図 3. Kubernetes によるコンテナオーケストレーション

コンテナベースシステムの Helm 構成管理

Helm は、複雑な Kubernetes システムを反復可能かつ管理可能な方法で記述できる便利な方法を提供するオープンソースプロジェクトです。Kubernetes 用のパッケージマネージャーと考えることができます。この設定は、「チャート」と呼ばれるドキュメントに記載されています。Helm チャートを使用すると、わずか数分で完全な NetIQ Access Manager システムを Kubernetes クラスターに導入できます。アップデートも迅速かつ簡単にできます。Helm は、以前の設定へのロールバックもサポートしています。

Helm は、複雑な Kubernetes システムを反復可能かつ管理可能な方法で記述できる便利な方法を提供するオープンソースプロジェクトです。

システム導入プロセスを以下に示します。コンテナイメージは OpenText によって維持され、イメージレジストリを介して提供されます。NetIQ Access Manager には、特定の要件に合わせてカスタマイズできる Helm チャートテンプレートが用意されています。このチャートでは、導入するコンテナの数、導入先、ネットワーク構成、監視、フォールトトレランス、永続ストレージなどが定義されます。Helm チャートを Kubernetes に適用して NetIQ Access Manager を導入します。Kubernetes は指定されたバージョンのコンテナイメージをリポジトリからダウンロードし、そのイメージに基づいてコンテナを作成します。コンテナは初期化スクリプトを実行し、NetIQ Access Manager の初期設定を完了します。このプロセスは、わずか数分で完了します。

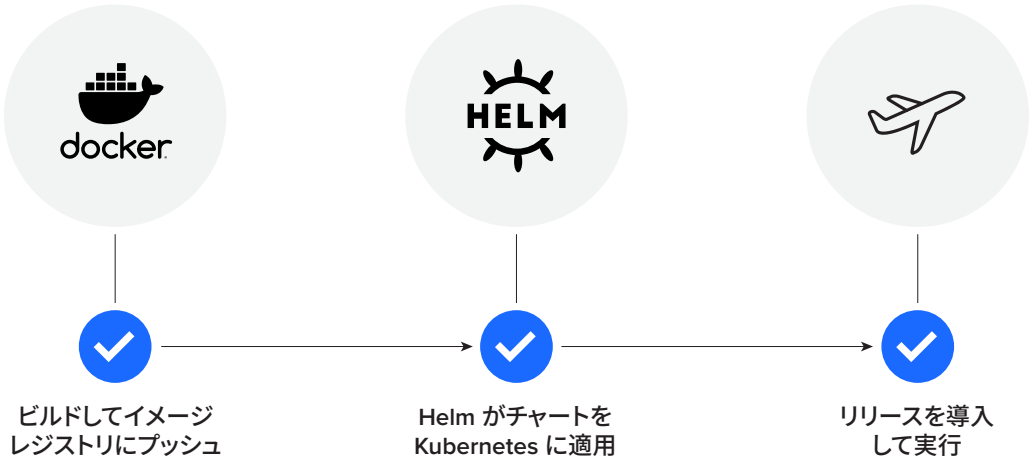

図 4. コンテナベースの導入

複雑な NetIQ Access Manager システムの移行は難しく見えるかもしれませんが、適切に計画すれば、プロセスの単純化で 10 倍のメリットがあります。

NetIQ Access Managerの最新バージョンへのアップデートも簡単です。使用したいバージョンを参照するよう Helm チャートを更新し、更新したチャートを適用します。このプロセスも数分で完了します。

こうした最先端のテクノロジーはどれも複雑で高価なものだと思われるかもしれません。ありがたいことに、それを自分で構築して維持する必要はないのです。すべての主要なパブリッククラウドプロバイダーで、フル機能の Kubernetes サービスが利用できます。プライベートクラウドベンダーもすべて Kubernetes をサポートしています。Amazon Elastic Kubernetes Service (EKS) はマネージド Kubernetes サービスで、独自の Kubernetes クラスターをインストールして操作する必要はなく、Kubernetes ベースのシステムを簡単に AWS に導入できます。Microsoft も Azure Kubernetes Services (AKS) で同様のサービスを提供しています。NetIQ Access Manager の導入は、両方のシステムで文書化、サポート、および認定されています。

NetIQ Access Manager の移行

複雑な NetIQ Access Manager システムの移行は難しく見えるかもしれません。適切に計画すればプロセスを単純化できます。移行をうまく進めるには、次のようにします。

1. 現在の実装を評価する。
2. 将来の要件を特定する。
3. クラウド導入プラットフォームを選択する。
4. 移行アプローチを決定する。
5. 運用モデルを特定し、必要なトレーニングを実施する。
6. 実装を設計する。

7. テスト環境を導入する。
8. 機能と性能をテストする。
9. 本番システムを導入する。
10. 本番システムのテストと検証を実施する。
11. アプリケーションと機能を移行する。

現在の NetIQ Access Manager の実装の評価
最初のステップは、現在のシステムの動作と構成を理解することです。理想的には、必要な情報のほとんどが、通常の運用文書としてあらかじめ用意されているでしょう。ただし、そのような状況はまれで、必要なドキュメントを適切に管理している組織はほとんどありません。適切なドキュメントがあっても、アクセス管理の統合はプロジェクトとして実施されることが多く、プロジェクトが終了すると、チームは解散します。統合の変更やテストの担当者、または担当できる人がいない可能性があります。

以下の作業が必要です。

- NetIQ Access Manager コンポーネントで使用しているものと使用していないものを確認する。
- 外部アイデンティティプロバイダーへのフェデレーションが使用されているかどうかを確認する。
  - どのプロトコルか
  - 組織内の誰が各アイデンティティプロバイダーとの関係を「所有」しているか
  - 各プロバイダーのテストプロセスはどのようなものか
  - アイデンティティプロバイダーの設定を変更できるのは誰か、必要な変更を実装するためのプロセスは何か
- フェデレーションサービスプロバイダーのカタログを作成する。
  - サービスプロバイダーを変更する権利を持つのは誰か
  - 構成変更を実装するために契約サポートが必要か
  - 変更を実施するために必要なプロセスとリードタイムはどのようなものか
  - 組織内でのアプリケーションの責任者は誰で、「ビジネスオーナー」は誰か
  - 統合をテストする能力と責任を持っているのは誰で、文書化されたテストプロセスがあるか

将来の要件の特定
現在のシステムの理解が十分になれば、次のアクティビティは、システム機能に必要な変更や発生する変更を特定することです。どのような新しい統合や機能が必要になるかだけでなく、どのような統合や機能を廃止できるかについても検討する必要があります。その例としては、フォーム入力やヘッダー挿入に基づくレガシー統合で、フェデレーションが可能になるものが挙げられます。また、従来のフェデレーションプロトコルを使用した統合で、最新のプロトコルをサポートできるものもあります。

次のような質問への回答が必要です。

- 今後どのようなフェデレーションプロトコルが必要になるか？一部の新しい統合では、OAuth や OpenID Connect が必要になる可能性があります。
- Access Gateway プロキシは必要になるか？プロキシを使用するとセキュリティと機能が強化されますが、完全なフェデレーションモデルに移行すると運用を簡素化できます。
- 新しいアプリケーションはあるか？
- システムの負荷に大きな変化が予想されるか？

どのような新しい統合や機能が必要になるかだけでなく、どのような統合や機能を廃止できるかについても検討する必要があります。

クラウド導入プラットフォームの選択

クラウド導入プラットフォームを決定するための最初のステップは、レガシーモデルとコンテナベースモデルのいずれかに決定することです。OpenText では、可能な限りコンテナを使用することをお勧めしています。レガシーモデルを使用する場合は、Azure と AWS のどちらも仮想マシンの導入用の認定プラットフォームであり、プロセスが文書化されています。

コンテナを選択した場合には、次のオプションがあります。

- Amazon Elastic Kubernetes Services (EKS)
- Azure Kubernetes Services (AKS)
- 別の Kubernetes プラットフォーム (クラウドベースの仮想マシン上に独自のクラスターインフラストラクチャを作成する場合を含む)

EKS と AKS はどちらも OpenText でフルにサポートされ、認定されています。他の Kubernetes インフラストラクチャを使用することもでき、OpenText はオーケストレーションをベストエフォートでサポートします。

運用モデルの特定とトレーニングの実施

アクセス管理システムの運用は非常に困難な場合があります。組織内または組織外の他のグループが所有および管理するアプリケーションを扱う場合には、特に困難さが増します。クラウドベースの導入に移行すると、新しいテクノロジーがそこに加わります。こうした課題には、移行計画の一環として対処することが必要です。組織内で新しいシステムをサポートする担当者を特定し、その担当者に知識を与えトレーニングを実施する必要があります。OpenText のトレーニングとプロフェッショナルサービスは、標準的なトレーニングプログラムにもカスタマイズされたトレーニングプログラムにも役立ちます。

移行アプローチの選択

このセクションでは、既存のアクセス管理システムから新しいシステムに移行する際のオプションについて検討します。ここで説明する概念は、あらゆる種類のアクセス管理システムから別のシステムへの移行に適用可能です。しかし、主眼を置いているのは、既存の NetIQ Access Manager システムから新しいクラウドベースの NetIQ Access Manager システムへの移行です。ユーザーデータと資格情報は、既存のデータストアを引き続き使用するか、既存のデータストアを新しいデータストアと同期するかのいずれかの手段で維持されると仮定しています。これが不可能な場合は、OpenText™ プロフェッショナルサービスにお問い合わせください。

以下は、NetIQ Access Manager システムの概念図で、移行を計画する際に考慮すべき要素を示しています。

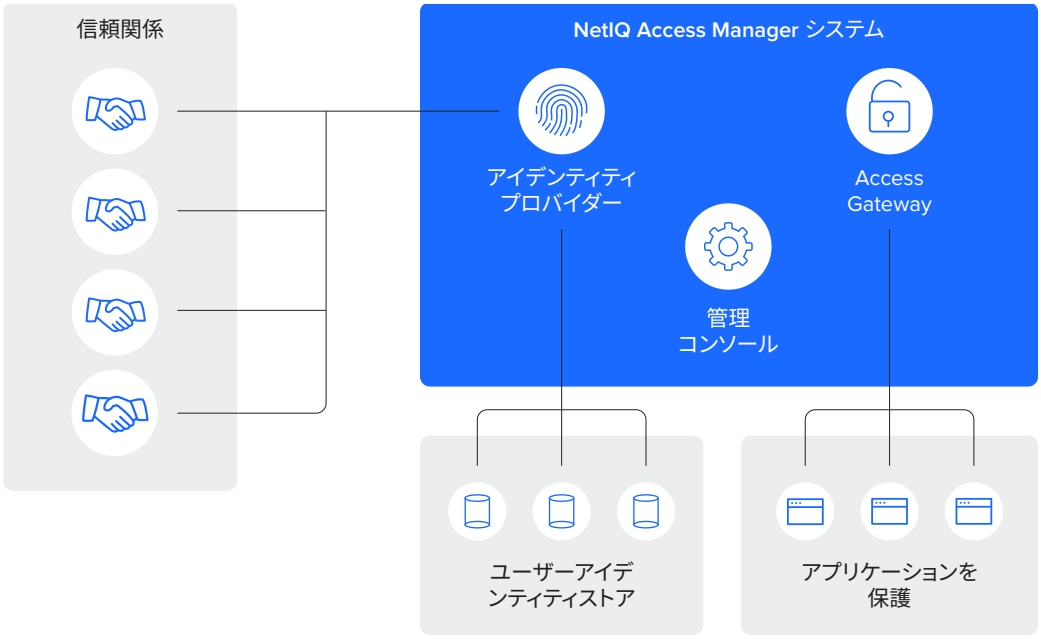

図 5. 移行の計画で考慮すべき要素

あるアクセス管理システムから別のシステムに移行する際の最大の課題は、移行中に既存の信頼関係を維持することです。システムには何百、場合によっては何千もの信頼関係が存在します。スケールが大きいと、急な一括切替は不可能です。関係の数がかなり少ない場合でも、一般的には変更とテストに外部の関係者や組織内の他の部門との調整が必要になるため、移行に影響が及ぶ可能性があります。

NetIQ Access Manager システムの「識別情報」をどのように変更しても、信頼関係に影響が発生します。システム内の識別情報が信頼関係にどのようにリンクするかは、フェデレーションプロトコルによって異なります。SAML の場合、重要な要素は SAML エンティティ ID、サービスエンドポイント URL、署名や暗号化に使用される証明書です。これらの要素のいずれかを変更すると、アイデンティティプロバイダーと信頼関係を持つすべてのサービスプロバイダーの設定を変更する必要があります。

システム中の識別情報に影響を与えるものを何も変更しないことで、問題を回避するのが最善であると、まず思いつくかもしれません。これは実行可能なアプローチですが、インフラストラクチャの違いや法的制限などの要因によって不可能な場合があります。既存のシステム識別情報を維持できる状況であっても、大規模で複雑なシステムですべてが正常に機能することをテストして検証するにはどうすればよいでしょうか？システム識別情報が維持される場合でも、突然の一括切替はほとんどの場合許容されるオプションではありません。

Access Gateway で保護されているアプリケーションの移行も考慮すべき事項です。アプリケーションの数と、各アプリケーションが適切に機能し続けることの検証の複雑さによって、手早く一括切替することが可能かどうか、あるいはより系統的なアプローチをとる必要があるかどうかが決まります。

移行アプローチを決定する際にさらに検討すべき要因は、リスク許容度と許容可能なダウンタイムです。一定のダウンタイムが許容できる状況は、ダウンタイムが許容できない状況とは大きく異なります。各アプローチによって、リスク管理に使用できるオプションが異なります。移行プロセスを小規模なアクティビティに分割して、各アクティビティの範囲と影響を低減できるものもあります。

お客様のニーズに最適な移行アプローチを選択してください。

- 現在のエンティティ ID を維持する新しいシステムへの完全な一括切替
- 全く別な新しいシステムへの完全な一括切替
- 統合されていない段階的な移行
- スパンクラスター移行

移行アプローチに使用できるオプションは次のとおりです。

1. 現在のエンティティ ID を維持する新しいシステムへの完全な一括切替。
2. 全く別な新しいシステムへの完全な一括切替。
3. 統合されていない段階的な移行 (移行中にシステム間のシングルサインオン (SSO) を行わない)。
4. 統合された段階的移行 (移行中にシステム間で SSO を実施)。
5. スパンクラスター移行 (現在のエンティティ ID を維持)。

アプローチ 1：現在のエンティティ ID を維持する新しいシステムへの完全な一括切替
このアプローチでは、アイデンティティプロバイダーは現在のエンティティ ID を維持します。信頼できるパートナーの設定を更新する必要は生じません。各信頼関係のテストを調整する必要はありますが、問題が発生するリスクは非常に低くなります。Access Gateway で保護されているアプリケーションは、ホストファイルまたは代替 DNS サーバーを使用して、移行前に完全にテストできます。テストが完了したら、新しいサービスを指すように DNS を変更します。この一般的なアプローチは、一括切替の前にすべての統合をテストすることが可能で、システム検証が管理可能な規模である場合にうまく機能します。このアプローチの問題点は、一発勝負だということです。統合のどこかで問題が発生すると、移行全体をロールバックする必要があります。ありがたいことに、ロールバックするには、トラフィックを古いシステムに戻すだけで済みます。

このオプションを使用すると、既存のシステムのアップグレードの必要がなく、現在の状態のままにしておくことができます。これにより、全体に必要な労力を軽減できます。このアプローチは、既存のシステムが NetIQ Access Manager ではなく、フェデレーション標準との適合性が高い場合にも適しています。

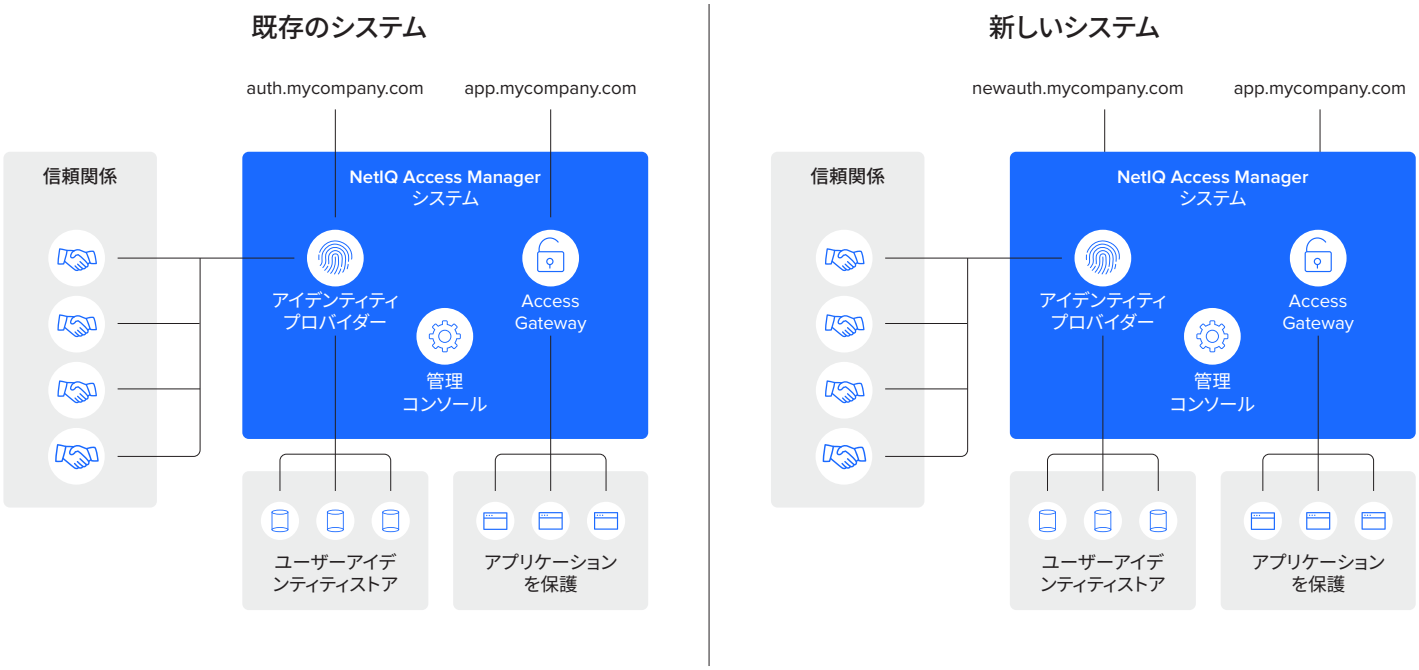

図 6. アプローチ 1：アイデンティティエンティティを維持する新しいシステムへの一括切替

実装プロセスは以下のとおりです。

- 新しいシステムをセットアップし、アイデンティティプロバイダーを同じ DNS 名で構成する。
- 新しい Access Gateway 上で、古いシステム上のエントリを複製するようにリバースプロキシを設定する。
- ホストファイルエントリを使用して、新しいシステムを可能な限り完全にテストする。
- すべてのフェデレーションパートナーと Access Gateway クラスターをまとめて一括切替する。この場合、信頼できるパートナーの設定を変更する必要はありません。
- すべてのアプリケーションとフェデレーションをテストする。

アプローチ 2：全く別な新しいシステムへの完全な一括切替

このアプローチにより、アイデンティティプロバイダーはまったく新しいエンティティになります。これは、フェデレーションが使用されていない場合、またはすべてのフェデレーションパートナーの設定を変更するための完全な制御権がある場合にのみ有効です。サードパーティプロバイダーとの間の複数のフェデレーションを更新するために必要な調整を、1 つの変更ウィンドウ内で取ることはほとんど不可能です。一括切替前にフェデレーションアプリケーションを適切にテストすることは困難で、一括切替中に問題が発生した場合にロールバックすると問題につながる可能性があります。それでも、このアプローチは、必要な範囲の制御権を組織が持っている少数のアプリケーションをサポートするシステムでは有効なオプションです。また、既存のシステムをアップグレードする必要もありません。このアプローチをここで説明するのは、全オプションを網羅するため、また後で説明する管理が容易なアプローチの基礎であるからです。

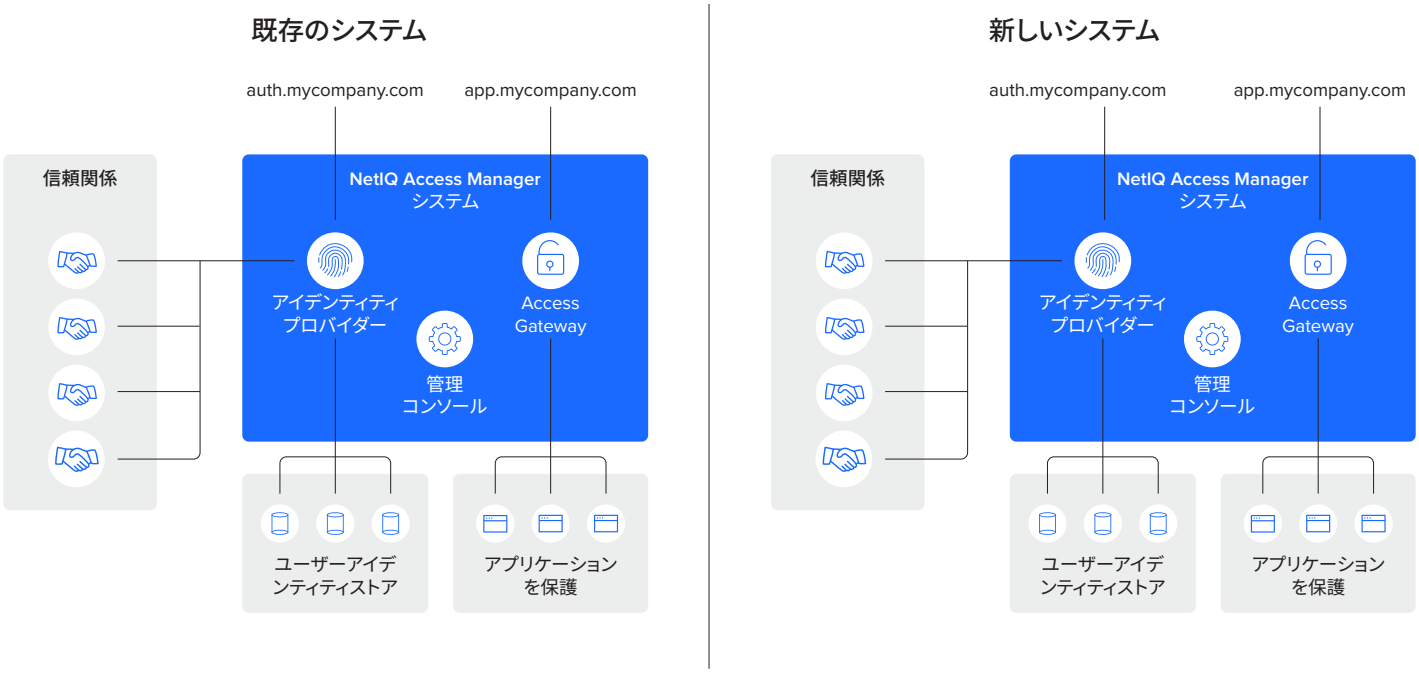

図 7. アプローチ 2：全く別な新しいシステムへの完全な一括切替

実装プロセスは以下のとおりです。

- 新しいシステムをセットアップし、アイデンティティプロバイダーを新しい DNS 名で設定する。
- 新しい Access Gateway 上で、古いシステム上のエントリを複製するようにリバースプロキシを設定する。
- ホストファイルエントリを使用して、新しいシステムを可能な限り完全にテストする。
- すべてのフェデレーションパートナーと Access Gateway クラスターをまとめて一括切替する。
- すべてのアプリケーションとフェデレーションを検証する。

アプローチ 3：統合されていない段階的な移行

このアプローチはアプローチ 2 の拡張版です。システム全体、すべての信頼関係、保護されたすべてのアプリケーションを一度に移行するのではなく、時間をかけて段階的に移行します。アプローチ 3 は、各移行アクティビティの範囲とリスクを大幅に削減します。しかし、ユーザー体験に影響を及ぼす可能性があり、それは多くの場合に受け入れられないようなものです。ユーザーが両方のシステムにログインする必要がある場合があり、アプリケーション間の相互依存関係のために移行アクティビティの範囲が必要以上に大きくなる場合があります。このオプションは、両方のシステムですべての移行に必要な操作を実施することに問題がなければ、多数の信頼関係がある場合にうまく機能します。

実装プロセスは以下のとおりです。

- 新しいシステムをセットアップし、アイデンティティプロバイダーを新しい DNS 名で設定する。
- 新しい Access Gateway 上で、古いシステム上のエントリを複製するようにリバースプロキシを設定する。
- ホストファイルエントリを使用して、新しいシステムを可能な限り完全にテストする。
- 適切な変更ウィンドウ内で合わせて移行および検証できる保護されたアプリケーションのセットを特定する。これは、ドメイン名全体を一度に移行するために、リバースプロキシレベルで行う場合がよくあります。
- 各信頼関係を新しいシステムに系統的に移行する。

アプローチ 4：統合された段階的移行

このアプローチは、アプローチ 3 と基本的には同じで、既存のシステムと新しいシステムとの間に信頼関係を確立する点が異なります。これにより、ユーザーのシングルサインオンエクスペリエンスを維持できます。実際には印象よりも複雑になる可能性があり、高度なフェデレーションの専門知識が必要です。信頼できる各パートナーおよび各アプリケーションについて、考えられるすべてのアクセスシナリオを評価する必要があります。信頼関係チェーンを介したユーザーを認証ができなくなることが例外的に発生します。この方法は、両方のシステムを長期間並行して運用する場合に最も適しています。

実装プロセスは以下のとおりです。

- 新しいシステムをセットアップし、アイデンティティプロバイダーを新しい DNS 名で設定する。
- 既存のシステムと新しいシステムの間に信頼関係を実装する。
- 新しい Access Gateway 上で、古いシステム上のエントリを複製するようにリバースプロキシを設定する。
- ホストファイルエントリを使用して、新しいシステムを可能な限り完全にテストする。各アクセスユースケースが、システム間の信頼関係を通じてスムーズに機能しているか、評価する必要があります。
- 適切な変更ウィンドウ内で合わせて移行および検証できる保護されたアプリケーションのセットを特定する。これは、ドメイン名全体を一度に移行するために、リバースプロキシレベルで行う場合がよくあります。
- 各信頼関係を新しいシステムに系統的に移行する。

アプローチ 5：スパンクラスター移行

このアプローチは、既存の NetIQ Access Manager システムから新しい NetIQ Access Manager システムに移行する場合に固有のものです。現在のシステムを最新のバージョンおよびパッチレベルにアップグレードしてから続行することが必要です。この場合、クリーンなシステムで最初からやり直す必要はなく、システム識別情報に変更はありません。既存のクラスターノードを新しいクラスターノードに置き換えるだけです。このシナリオで複雑な点は、適切なネットワーク接続を確保し、グローバルとローカルの両方のロードバランシングを管理することです。

多くの場合、レガシーインフラストラクチャと新しいインフラストラクチャを並行して運用できます。この「ハイブリッド」モデルでは、耐障害性が強化され、完全なクラウドベースの導入が容易になります。下の図は、AWS に導入されたクラウドコンポーネントを含むハイブリッドインフラストラクチャを示しています。AWS ノードは、コンテナとしても仮想マシンとしても導入できます。

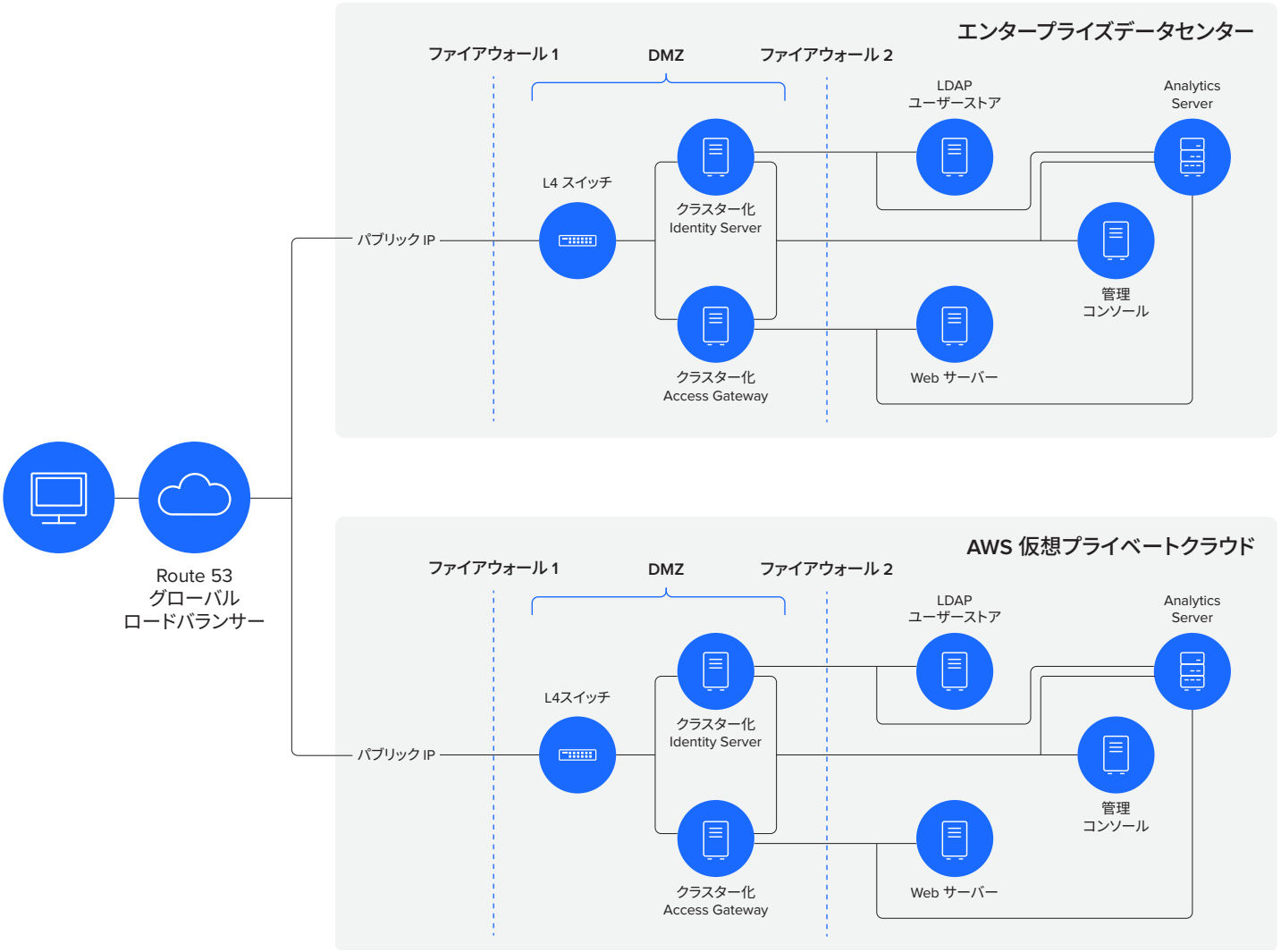

図 8. ハイブリッドインフラストラクチャモデル

実装プロセスは以下のとおりです。

- 新しい環境と既存の環境の間にネットワーク接続が存在することを確認する。各コンポーネントに必要な接続については、NetIQ Access Manager のマニュアルを参照してください。オンプレミスアプリケーションがクラウドベースの Access Gateway を介してプロキシされる場合は、ネットワークパフォーマンス要件に特に注意してください。

- 新しい環境で新しい Identity Server ノードと Access Gateway ノードを設定する。
- 新しい環境で新しいセカンダリ管理コンソールをセットアップする。
- 既存のクラスター構成に新しいノードを追加する。
- ホストファイルエントリを使用して、新しいノードを可能な限り完全にテストする。
- 可能であれば、新しいノードをロードバランシングスキームに追加する。これが不可能な場合は、各 DNS 名を個別に移行できます。
- 各アプリケーションとフェデレーション関係をテストする (IP ベースのホワイトリストに起因する問題に注意)。
- 必要に応じて、レガシーなノードとインフラストラクチャの運用を停止する。

システム実装の設計

移行アプローチを選択したら、次のステップでは実装を設計します。幸いなことに、NetIQ Access Manager では、このような作業の多くを実施済みの Helm チャートテンプレートが提供されています。最初のステップは、ネットワークとフォールトトレランスのインフラストラクチャを決定することです。EKS にも AKS にも、次のようなガイドがあります。

- **https://docs.aws.amazon.com/eks/latest/userguide/eks-networking.html**
- **https://aws.amazon.com/quickstart/architecture/amazon-eks/**
- **https://docs.microsoft.com/en-us/azure/aks/concepts-network**

本番環境では、Kubernetes クラスターは少なくとも 2 つのワーカーノードで構成されている必要があります。この最小限の構成では、管理コンソール、Identity Server、Access Gateway のコンテナが 2 つのノードのそれぞれで実行されるように構成します。負荷とパフォーマンスの要件に応じて、Identity Server と Access Gateway の追加インスタンスを実行するためのノードを追加できます。一般的には、現在の NetIQ Access Manager クラスターと同じ数のインスタンスが必要になると想定できます。4 台の Identity Server と 6 台の Access Gateway がある場合は、同じ数のコンテナが必要になるでしょう。1 つのノードで Identity Server と Access Gateway の両方をホストできるため、実際に必要なワーカーノードの合計数は少ない場合があります。

次のステップでは、どのようにリクエストをコンテナにルーティングするかを決定します。EKS では、他のパブリック IP と同様に、パブリック IP ルーティングトラフィックを割り当てることができます。ただし、この場合は柔軟性が制限されます。もう 1 つの方法は、Ingress と呼ばれる Kubernetes サービスを使用することです。Ingress サービスは、基本的にアプリケーションレベルのファイアウォールおよびロードバランサーであり、コンテナのステータスに基づいてトラフィックをルーティングできます。AKS では、パブリック IP アドレスのオプションがないため、Ingress サービスを使用する必要があります。さらに、EKS と AKS のどちらにも、複数のデータセンターまたは可用性ゾーンにトラフィックをルーティングできるグローバルロードバランシングのオプションがあります。

各コンテナには、設定ファイルとログ用の永続ストレージが必要です。このストレージは、ワーカーノードに対してローカルなものでも、Kubernetes ボリュームドライバーで提供されるものでも構いません。EKS と AKS のどちらでも、ストレージを仮想化するオプションが提供されており、特定のノードに依存することなく簡単に保守できます。ノード間通信は IP アドレスに基づいているため、ストレージと IP アドレスを新しいワーカーノードに移動するよりも、新しいインスタンスをスピンアップする方が簡単な場合があります。NetIQ Access Manager のネイティブクラスターアーキテクチャでは、単一のノードに依存することは全くありません。

完了すると、設計は Helm チャートとして表現され、Kubernetes クラスターに適用できるようになります。グローバルロードバランシング構成など、Kubernetes が管理しないインフラストラクチャ要素が一部存在する可能性があることに注意してください。これらの要素は別途設定が必要です。EKS では、これらの要素のほとんどを AWS CloudFormation テンプレートを使用して指定できます。AKS にも同様の機能があります。

導入、テスト、移行

導入は非常に簡単です。EKS で必要な手順の概要は次のとおりです。

1. AWS アカウントを持っていない場合は、作成する。
2. システムの導入で超過するアカウントのリソース制限があるかどうかを確認する。VPC、セキュリティグループ、IAM ロール、自動スケーリンググループ、インスタンスの数には制限があります。アカウントでさほど AWS リソースを使用していない場合は、これは問題になりません。
3. NetIQ Access Manager 用の新しい VPC を作成する (既存の VPC を使用可能)。
4. VPC に EKS クラスターを作成する。
5. クラスターのワーカーノードを作成する。
6. 永続ストレージボリュームを作成する。
7. システムのアクセス管理のためのセキュリティグループと IAM ロールを作成する。
8. AWS クラスターと接続するためにローカルワークステーションで kubectl 管理ユーティリティを構成する。
9. ワークステーションに Helm をインストールする。
10. Helm チャートを適用する。

Helm チャートが適用されると、コンテナイメージが NetIQ Access Manager リポジトリからダウンロードされ、ワーカーノードで実行されます。このプロセスには約 10 分かかります。このプロセスが完了すると、NetIQ Access Manager システム全体を構成できるようになります。

最終的な構成に進む前に、Kubernetes での NetIQ Access Manager システムの操作に慣れておく必要があります。システムの破壊と再構築、コンテナの追加と削除、NetIQ Access Manager イメージのアップグレードについて試しておいてください。これらすべてのプロセスを使いこなせることが必要です。この時点で、自動スケーリング、ロードバランシング、フォールトトレランスもテストする必要があります。

インフラストラクチャが安定していることを確認したら、Identity Server と Access Gateway の構成を完全に実装する必要があります。この時点では、DNS はまだ既存のシステムにトラフィックを向けていますが、ホストファイルまたは代替 DNS サーバーを使用してテストできます。ホストファイルを使用する場合は、コンテナからの DNS 解決によって正しいテストアドレスが得られていることも確認する必要があります。

構成を検証できたら、負荷テストを実行して、新しいシステムが既存のシステムと同等以上のパフォーマンスを発揮できることを確認する必要があります。継続的なキャパシティプランニングに使用できるパフォーマンスベースラインを確立することも必要です。

最後のステップは、選択した移行アプローチに基づいてアプリケーションと統合を移行することです。

デジタルトランスフォーメーションが組織のプロセスに深く入り込み、ビジネス全体で拡大するにつれて、NetIQ Access Manager のお客様の中で、基盤となるインフラストラクチャの管理をクラウドに移行するケースがますます増えています。NetIQ Access Manager の障害復旧環境の維持にかかるコストとオーバーヘッドを軽減し、サービスパフォーマンスの予測性を高めることが可能になります。このポジションペーパーによって、クラウドへのアクセス管理の移行がより簡単になることを願っています。

OpenText について

このたび、OpenText による、CyberRes を含む Micro Focus の買収が完了しました。両社の専門知識の融合によって、セキュリティ製品 / サービスの提供が拡張され、権限とアクセスの制御の自動化を通じて、アプリケーション、データ、リソースへの適切なアクセスを確保することにより、お客様の機密情報の保護を支援します。NetIQ Identity and Access Management は OpenText Cybersecurity の一部であり、あらゆる規模の企業やパートナーに包括的なセキュリティソリューションを提供します。

お問い合わせ
www.opentext.com

opentext™ | Cybersecurity

OpenText Cybersecurity は、あらゆる規模の企業とパートナー様を対象に、包括的なセキュリティソリューションを提供しています。予防から検出、復旧対応、調査、コンプライアンスに至るエンドツーエンドの統合プラットフォームにより、包括的なセキュリティポートフォリオを通じてサイバーレジリエンスの構築をサポートします。コンテキストに基づくリアルタイムの脅威インテリジェンスから得られた実用的なインサイトを活用できるため、OpenText Cybersecurity のお客様は、優れた製品、コンプライアンスが確保されたエクスペリエンス、簡素化されたセキュリティというメリットによって、ビジネスリスクを管理できます。

762-JA0046-004 | O | 05/24 |