NGINX
Part of F5

Ingressコントローラーを選ぶ際の指針

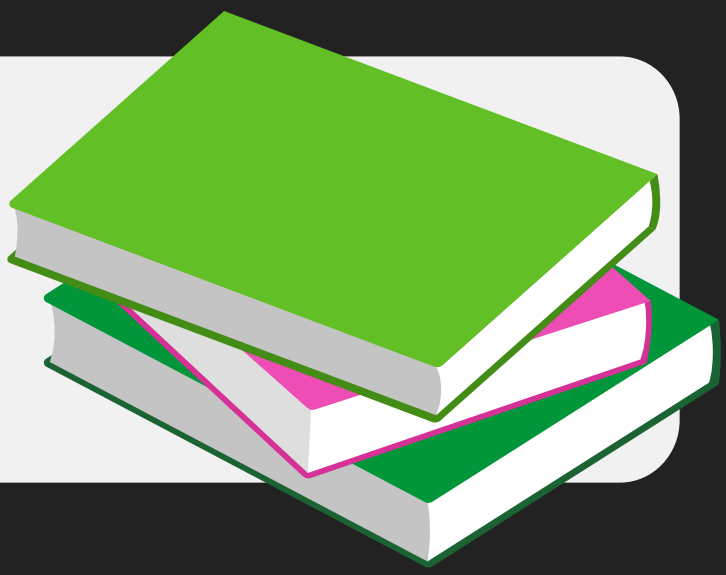
Kubernetesの「保証ある実現」を具体化しましょう

- Kubernetesを実現する上で重要なことは、**コスト削減を実現しながら、**優れたデジタルエクスペリエンスをより**迅速かつ安全に**提供可能とすることです。

- しかし、クラウド移行やマイクロサービスに慣れていたとしても、本番環境でのKubernetesシステムの運用は**簡単ではありません。**実際にKubernetesを使用し始めてみると、アプリケーションを**保護することや、把握し 常に適切であるか確認すること**が困難になることがあります。

- **Ingressコントローラー**はKubernetesスタックの中で最も**強力なツール**の1つであり、Kubernetes実現に最も重要なサポートをします。

Ingressコントローラーの基本と、**現在**および**将来**、必要となる機能とセキュリティを提供するための**賢い選択方法**について説明しましょう。



Ingressコントローラーの役割

Ingressコントローラーは、レイヤ4や7におけるトラフィックの出入り（ingress とegress）を管理する専用のロードバランサーです。

また、次の事項にも使用できます。

- トラフィックコントロール
- トラックスシェーピング
- モニタリングと可視化
- APIゲートウェイ
- 認証とSSO
- WAFの統合

モニタリングと可視化

Ingressコントローラーを使用すると、アプリとインフラストラクチャのパフォーマンスに影響を与える問題を把握し、トラフィックの急増がいつ発生するかを予測可能となります。

セキュリティ

Ingressコントローラーは、一元化された認証、シングルサインオン（SSO）、およびWeb アプリケーションファイアウォール（WAF）の理想的なポイントとして、不正または悪意のあるトラフィックから環境を保護します。

HACKERエアー

Ingress航空

Egress航空

Ingress
コントローラー

Ingress トラフィックはKubernetesクラスタに入ってくるトラフィックです。

IngressコントローラーはIngressトラフィックを受け入れ、それを変更（シェーピング）し、環境内で実行されているポッドに配信します。

Egress トラフィックはKubernetesクラスタから外部に出ていくトラフィックです。

Ingressコントローラーは、egressのルールを実装して、相互TLS（MTLS）によるセキュリティを強化、あるいは特定のポッドから特定の外部サービスへの発信トラフィックを制限します。

サービス
A

サービス
E

Kubernetes
インターナショナル
空港

サービス
B

サービス
D

サービス
C

Ingressコントローラーは、**サービス**の個々のポッドをモニタリングし、インテリジェントルーティングを保証し、リクエストが「ブラックホール化」されるのを防ぎます。

サービスメッシュ

サービス
E

サービスメッシュ

サービスメッシュは、East-West（サービス間）トラフィックをルーティングし、保護します。

サービスメッシュは、次を実装するために使用されます。

- エンドツーエンドの暗号化とmTLS
- オーケストレーション
- サービストラフィックの管理
- モニタリングと可視化

サービス
D

East-West
トラフィック

East-West（サービスツーサービス）トラフィックは、Kubernetesクラスタ内のサービス間を移動するトラフィックです。

Ingressコントローラーでは、East-Westのサービス間トラフィックを管理することができません。

アプリケーションやインフラストラクチャが成熟しサービス間トラフィックを管理する必要が生じると**サービスメッシュ**が必要になります。

Ingressコントローラーに
どのようにしてリソースを
準備しますか?

資本コストの
予算作成

時間コストの
予算作成

Ingressコントローラーのリスク

新しいツールを使用すると、メリットよりもリスクが高くなる場合があります。ここでは、ニーズに合っていないIngressコントローラーがもたらす可能性のある上位3つのリスクについて説明します。

01 複雑さ

マイクロサービスアーキテクチャの目的から逸脱していませんか?

複雑さは、コンテナの使用とデプロイメントにおいて、最重要課題の1つです。[1]

Ingressコントローラーを誤って使用すると、さらに複雑さが増し、デプロイメントを水平に拡張する能力が制限され、アプリのパフォーマンスに悪影響を与える可能性があります。

02 レイテンシー

Ingressコントローラーはアプリの速度を低下させてしまいますか?

組織はKubernetesを採用することで、新しいアプリをより迅速に展開できるようになります。[2]

しかし、Ingressコントローラーを使うことで、エラー、タイムアウト、およびリロードによるレイテンシーが生じる可能性があり、それがアプリの速度を低下させる場合があります。

03 セキュリティ

Ingressコントローラーはハッカーに扉を開いてしまう可能性がありますか?

組織の半数以上が、コンテナまたはKubernetesのセキュリティ上の懸念により、本番環境へのアプリケーションのデプロイメントを遅らせたり、遅くしたりしています。[3]

CVE（脆弱性）パッチの適用が遅いIngressコントローラーに注意し、パブリックなフォーラムからのサポートに依存しすぎないように注意してください。

[1] CNCF Survey 2020（CNCF 調査 2020）
[2] 2021 Kubernetes Adoption Survey
（2021 年 Kubernetes 導入に関する調査）
[3] Red Hat State of Kubernetes Security Report
（Red Hat - Kubernetes セキュリティの状況レポート）

Ingressコントローラーの将来性

Kubernetesを試し始めたばかりでも、いつかそれを本番環境に投入しようと熱望する可能性は十分にあります。

時間の経過と共に、ニーズが徐々に高まる可能性があるのは、主に4つの分野です。

01

インフラストラクチャ

ハイブリッドあるいはマルチクラウド環境でKubernetesを使用しますか？

組織が1つのタイプの環境に完全かつ永続的にコミットするということは稀です。インフラストラクチャに依存しないIngressコントローラーを最初から選択しておくと、すべての環境で同じツールを使用することが可能となります。

02

セキュリティ

Kubernetesを内部からどのように保護しますか？

Kubernetesアプリは、認証や承認を含むセキュリティがアプリに近い場合に最も良く保護されます。セキュリティ（認証、承認、DoS攻撃からの保護、Webアプリケーションファイアウォール）をIngressのポイントで一元管理することは、コストと効率の両方の観点から見ると、非常に理にかなったことです。

03

サポート

どの程度「独り立ち」できますか？

小規模なデプロイメントを実行している場合は、回避策とコミュニティのサポートに期待しても構いませんが、本番環境に移行する場合は持続可能ではありません。将来的にサポートを追加できるIngressコントローラーを選択するか、規模に応じてアップグレードできるサポートを選択してください。

04

マルチテナンシー

複数のチームとアプリがコンテナ環境を安全かつ安全に共有するにはどうすればよいか？

サービスやチームの規模と複雑さが増すと、最大限の効率性を実現するために、マルチテナンシーの利用が必要となります。一部のIngressコントローラーは、ロールベースアクセスコントロール（RBAC）の設定をサポートする複数の入力アドレス、クラス、ネームスペース、およびスコープ指定リソースなど、さまざまな機能と概念を使って、これらのクラスタの分割を可能とします。

オープンソースIngressコントローラー

ユーザーとボランティア開発者のコミュニティによって維持管理されていますが、専任のエンジニアリングチームがある場合もあります。

長所

オープンソースのIngressコントローラーが最適である主な理由

- ▲ **金銭的投資の必要がない（無料！）**
- ▲ **コミュニティ主導型**
- ▲ **高機能で高速**

理想的なケースとは…
Kubernetesを使い始めたばかり、またはテストや小さなプロジェクトを開始したばかり。

短所

オープンソースのIngressコントローラーが最適ではない主な理由

- ▼ **要する時間が多くその分コストがかかっている**
- ▼ **安定性や信頼性の保証はなくリスクを含む**
- ▼ **最小限のサポートまたはサポートがない**

短所にあるリスクを避けるためには、「デフォルト」または「商用版」のオプションを検討してください。

デフォルトのIngressコントローラー

Kubernetesの完全プラットフォームを提供（そして多くの場合、その管理をサポート）する企業が開発および保守を行います。

長所

デフォルトのIngressコントローラーが最適である主な理由

- ▲ **無料または低コスト**
- ▲ **高い信頼性**
- ▲ **サポートがある**

理想的なケースとは…
Kubernetesプラットフォームをすでに使用しているが、まだテストあるいは小さなプロジェクトを開始したばかり。

短所

デフォルトのIngressコントローラーが最適ではない主な理由

- ▼ **インフラストラクチャ・ロックイン**
- ▼ **ベーシックな機能**
- ▼ **予測不可能な時間や費用**
  スケールにはコストがかかる

短所にあるリスクを避けるためには、「デフォルト」または「商用版」のオプションを検討してください。

商用版Ingressコントローラー

大規模な本番環境の導入をサポートするように設計されたライセンス製品です。

長所

商用版Ingressコントローラーが
最適である主な理由

▲ **豊富な機能セット**

▲ **スケーラブルで時間短縮が可能**

▲ **高い信頼性とサポート**

理想的なケースとは…
管理の複雑さを軽減し、新製品や機能の市場
投入までの時間を短縮しましょう。

短所

商用版Ingressコントローラーが
最適ではない理由

▼ **機能分速度が遅くなる**

▼ **金銭的投資が必要となる**

これらの短所を補うためには、「オープン
ソース」または「デフォルト」のオプショ
ンを検討してください。

NGINX Ingress Controllerによるセキュリティとコンプライアンスの向上

NGINX Plusベースの製品では、アプリと顧客の安全性を確保するために5つの重要なユースケースに対応しています。

01 エッジの保護

02 認証と承認の一元化

03 エンドツーエンド暗号化の実装

04 タイムリーでプロアクティブなパッチ通知を入手

05 FIPSに準拠

ドイツの最大手の自動車メーカー「Audi」がいかにRed Hat OpenShiftアプリを保護したかについては、『**Audi Future-Proof Tech Vision と App Innovation with NGINX**（アウディの未来に向けた技術ビジョンとNGINXによるアプリのイノベーション）』の導入事例をご確認ください。

NGINX Ingress Controllerによる アプリケーションの パフォーマンス向上と回復力

NGINX PlusベースのIngress コントローラーでは、Kubernetesを安全に実現するために役立つ5つのユースケースに対応しています。

01 ライブモニタリングの実現

02 障害の検出と解決の迅速化

03 ゼロ・リスタートによる再設定

04 新機能と展開の徹底的なテスト

05 サポートのニーズを迅速に解決

ビジネステキストメッセージング会社の「Zipwhip」では、**NGINXを使ってAmazon EKSのセキュリティとトラフィックの可視化を強化**し、SaaSアプリのアップタイムを99.99%達成しています。是非、この方法について導入事例をご覧ください。

詳細を確認したいですか?

ブログ連載を読む

パート 1：要件の特定

解決したい問題や、時間、お金、またはその両方をリソースとするかなど、Ingressコントローラーの要件を明かにします。

パート 2：リスクと将来の保証

Ingressコントローラーの選択を誤ることにより生じる可能性のあるリスクと、選択を将来にわたって保証できる重要な要因を認識します。

パート3 ：オープンソース vs. デフォルト vs. 商用

オープンソース、デフォルト、商用版の3つのカテゴリの長所と短所を詳しく調査し、Ingressコントローラーをどれにするか絞り込みます。

パート4 ：NGINX Ingress Controllerのオプション

著者、開発理念、本番環境の準備状況、セキュリティ、サポートに基づいて、どの NGINX Ingressコントローラーが最適か判断してください。

NGINX
Part of F5

ユースケースについてはお問合せください。