まえがき

本書に込めた思い

　はじめまして。本書を手に取っていただきありがとうございます。本書は、⛵ Istio（イスティオ）の使い方を解説した書籍です。近年、クラウドネイティブ技術の普及に伴い、フルスクラッチなマイクロサービスアーキテクチャを構築する必要はなくなってきました。クラウドネイティブ技術を駆使することによって、ベンチャー企業から大企業までのさまざまな企業がマイクロサービスアーキテクチャを構築できるようになりました。その一方で、マイクロサービスアーキテクチャを導入してみたものの、さまざまな問題を抱える企業が増えました。そういった潮流の中で、マイクロサービスアーキテクチャの問題を解決する『サービスメッシュ』が注目されています。本書では、サービスメッシュを実装するクラウドネイティブ技術である『Istio』に焦点を当て、概念から効果的な活用方法までを解説しました。

　さて、読者の皆さまは、『Istio』に対して以下のような疑問を持っているのではないでしょうか。

- 『サービスメッシュ』『Istio』は聞いたことあるが何を実現する技術なのか
- Istioを現場で導入する予定はないけど、勉強したい
- Istioを現場で導入してみたいけど、どのような問題がある時に導入するべきのか
- 現場でKubernetesは導入しているが、Istioがなくても困ってないし、導入しなくてもいいのではないか
- 現場でIstioをすでに導入しているが、使いこなせている感覚がなく、より効果的に活用したい

実際、会社の同僚やエンジニア仲間からこのような疑問を聞くことが多いです。これらの疑問に答えるため、Istioの全ての設定を網羅的に解説する役割は公式ドキュメントに譲りつつ、本書ではIstioの恩恵と効果的な活用方法に焦点を当てることにしました。サービスメッシュやIstioを理解する上で、以下のような前提があります。

- Istioはマイクロサービスアーキテクチャの問題を解決するための技術であるため、マイクロサービスアーキテクチャの知識が必要
- Istioのロジックをマイクロサービスアーキテクチャのアプリケーション領域とインフラストラクチャ領域に横断しており、両方の領域の広い知識が必要
- Istioを効果的に導入するには、マイクロサービスアーキテクチャや関連するOSSの知識が必要
- IstioはEnvoyの複雑さを抽象化してくれてるとはいえ、それでもEnvoyの知識が必要

そのため、Istioだけでなく関連知識も本書で提供しています。さいごに、実際の現場で求められるIstioのプラクティスや起こる不具合は、現場の状況によってさまざまです。そのため、本書で網羅しきれていないプラクティスやIstioの不具合でわからないことがあれば、いずれの媒体（メールアドレス、各種SNSなど）からでも良いので、気兼ねなくご連絡ください。

長谷川広樹（@Hiroki__IT）

想定読者

本書は、以下の読者を想定しています。IstioはKuberenetes上にデプロイすることを前提としているため、Istioと並行して、Kubernetesも学習いただくとよいかもしれません。

- Istio、クラウドネイティブ技術、マイクロサービスアーキテクチャを学習中の方
- 現場のマイクロサービスアーキテクチャで問題を抱えており、解決方法を探している方
- Istioの導入は決まっているものの、まだ躊躇している方
- 今現在、すでにIstioを導入しており、より効果的に活用したいと思っている方

章構成

本書は、以下のロードマップでIstioを解説しきます。

- 01章：Istioの前提知識を解説
- 04章：Istioのリソース全体を一覧で解説
- 05-11章：Istioの各リソースを実践的に解説
- 12-13章：Istioの重要なプラクティスを解説
- 14-15章：Istioを非技術的に語る
- 付録01-03：Istioの補足的なプラクティスを解説

より具体的に、章の内容とサンプルコードを表で整理しました。サンプルコードで扱わないIstioの関連リソースについても、書籍内では紹介しています。

章	概要	サンプルコードURL	Istioの関連リソース
1	マイクロサービスアーキテクチャとサービスメッシュの概要を解説します。Istioは、マイクロサービスアーキテクチャのデザインパターンの一つである『サービスメッシュ』を実装する技術です。その前提知識を解説します。	https://github.com/hiroki-it/istio-demo/tree/main/chapter-01	-
2	Istioの概要、アーキテクチャの種類、サイドカーモードを	https://github.com/hiroki-it/istio-	・ConfigMap（MeshConfig）

章	概要	サンプルコードURL	Istioの関連リソース
	構成する『コントロールプレーン』『データプレーン』の仕組みを解説します。サンプルコードでは、インストール方法の一つであるhelmfileを導入して、マイクロサービスアーキテクチャにIstioを導入します。	demo/tree/main/chapter-02	・DestinationRule ・Gateway Resource Annotation ・ServiceEntry ・VirtualService
3	Istioのコントロールプレーンの設定を一覧で解説します。各リソースの具体的な使い方やプラクティスは5章以降で解説します。	-	-
4	Istioのリソースの設定を一覧で解説します。各リソースの具体的な使い方やプラクティスは5章以降で解説します。	-	-
5	まず、マイクロサービスアーキテクチャで要求されるL4／L7トラフィック管理を解説します。その後、Istioサイドカーモードによる L4／L7トラフィック管理 （サービス検出、ロードバランシング）を実践します。	https://github.com/hiroki-it/istio-demo/tree/main/chapter-05	・ConfigMap（MeshConfig） ・DestinationRule ・Gateway Resource Annotation ・ServiceEntry ・VirtualService
6	まず、マイクロサービスアーキテクチャで要求される証明書管理を解説します。その後、Istioサイドカーモードによる証明書管理 （証明書作成、証明書ローテーション）を実践します。	https://github.com/hiroki-it/istio-demo/tree/main/chapter-06	・ConfigMap（MeshConfig） ・DestinationRule ・Gateway ・PeerAuthentication Resource Annotation ・ServiceEntry ・VirtualService
7	まず、マイクロサービスアーキテクチャで要求される認証／認可を解説します。その後、Istioサイドカーモードによる認証（特にユーザーの有効性検証）と認可（Enforcement、Decision）を実践します。	https://github.com/hiroki-it/istio-demo/tree/main/chapter-07	・AuthorizationPolicy ・ConfigMap（MeshConfig） ・DestinationRule ・Gateway ・RequestAuthentication ・PeerAuthentication Resource Annotation

章	概要	サンプルコードURL	Istioの関連リソース
			・ServiceEntry ・VirtualService
8	まず、マイクロサービスアーキテクチャで要求されるオブザーバビリティーを解説します。その後、Istioサイドカーモードによるテレメトリー作成を解説します。	https://github.com/hiroki-it/istio-demo/tree/main/chapter-08	・AuthorizationPolicy ・ConfigMap（MeshConfig） ・DestinationRule ・Gateway ・RequestAuthentication ・PeerAuthentication Resource Annotation ・ServiceEntry ・Telemetry ・VirtualService
9	まず、マイクロサービスアーキテクチャで要求されるブラックボックステストを解説します。その後、Istioサイドカーモードによるブラックボックステストのフォールトインジェクションを解説します。	https://github.com/hiroki-it/istio-demo/tree/main/chapter-09	・AuthorizationPolicy ・ConfigMap（MeshConfig） ・DestinationRule ・Gateway ・RequestAuthentication ・PeerAuthentication ・ServiceEntry ・VirtualService
10	まず、マイクロサービスアーキテクチャで要求される回復性管理を解説します。その後、Istioサイドカーモードによる回復性管理 （タイムアウト、リトライ、サーキットブレイカー）を解説します。	https://github.com/hiroki-it/istio-demo/tree/main/chapter-10	・AuthorizationPolicy ・ConfigMap（MeshConfig） ・DestinationRule ・Gateway ・RequestAuthentication ・PeerAuthentication Resource Annotation ・ServiceEntry ・VirtualService
11	IstioアンビエントモードによるL4／L7トラフィック管理を解説します。	https://github.com/hiroki-it/istio-demo/tree/main/chapter-11	・ConfigMap（MeshConfig） ・DestinationRule ・Gateway Resource Annotation ・ServiceEntry ・VirtualService

章	概要	サンプルコードURL	Istioの関連リソース
12	Istioサイドカーモードの設計プラクティスを紹介します。	-	-
13	Istioと組織論の関連性を解説します。	-	-
14	Istioのトラブルシューティング方法を解説します。	-	-
15	サービスメッシュが今後どうなっていくかを解説します。	-	-
付録1	Istioサイドカーモードのアップグレード手順を解説します。	https://github.com/hiroki-it/istio-demo/tree/main/chapter-extra-01	・ConfigMap（MeshConfig） ・DestinationRule ・Gateway Resource Annotation ・ServiceEntry ・VirtualService
付録2	Gateway APIを使用したIstioサイドカーモードを解説します。	https://github.com/hiroki-it/istio-demo/tree/main/chapter-extra-02	・ConfigMap（MeshConfig） ・DestinationRule ・Gateway Resource Annotation ・ServiceEntry ・VirtualService
付録3	ネイティブサイドカーを使用したIstioサイドカーモードを解説します。	https://github.com/hiroki-it/istio-demo/tree/main/chapter-extra-03	・ConfigMap（MeshConfig） ・DestinationRule ・Gateway Resource Annotation ・ServiceEntry ・VirtualService

サンプルコード

リポジトリ

本書のサンプルコードは以下のGitHubリポジトリで公開しています。

https://github.com/hiroki-it/istio-demo

リポジトリをクローンすることで、本書で解説するIstioの機能を実際に試すことができます。リポジトリは、以下のディレクトリ構成からなります。Istioが最初に登場するchapter-02ディレクトリに関して、各ディレクトリやファイルの役割を解説しました。

```
istio-demo/
├── README.md # Minikubeクラスターのセットアップ方法
├── bookinfo-app/ # BookinfoアプリケーションのHelm Chart
├── chapter-01/ # 01章に関するHelm Chart
│
├── chapter-02/ # 02章 〃
│   ├── README.md # 02章のセットアップ方法
│   ├── bookinfo-app # Bookinfoアプリケーションをサービスメッシュに登録するための
Istioリソース
│   ├── istio # Istiod、Istio Ingress Gateway、Istio Egress Gateway
│   ├── kiali # Kiali
│   ├── metrics-server # metrics-server
│   ├── prometheus # Prometheus
│   ├── setup.sh # READMEに記載の内容を一度でセットアップするためのシェルスクリプト
│   └── shared # ツール間で共有するリソース
│
├── chapter-05/ # 05章 〃
├── chapter-06/ # 06章 〃
├── chapter-07/ # 07章 〃
├── chapter-08/ # 08章 〃
├── chapter-09/ # 09章 〃
├── chapter-10/ # 10章 〃
├── chapter-11/ # 11章 〃
├── chapter-extra-01/ # 付録01章 〃
├── chapter-extra-02/ # 付録02章 〃
├── chapter-extra-03/ # 付録03章 〃
├── databases/ # Minikubeクラスター外のDB
├── images/ # READMEに使用する画像
├── mise.toml # 各ツール
└── setup.sh #
```

マシン環境

本書のサンプルコードを快適に実行するためには、以下のマシン環境を推奨します。

- CPU：8コア以上
- メモリ：16GB以上

サンプルコードのマイクロサービスアーキテクチャには、以下のマイクロサービスアプリケーションやIstioの他にも、さまざまなOSが含まれます。CPUの割り当てが不十分な場合、コンテナの性能が低下してしまいます。また、メモリの場合、Out Of Memoryによってコンテナが強制的に終了しPodが再起動してしまいます。できる限り、推奨環境を満たすマシンをご用意ください。

IstioとKialiのバージョン

本書で扱うIstioのバージョンは、<後で書く>とします。Istioの特定のバージョン以降、以下が異なる可能性があることに注意してください。

- API Versionが廃止されている
- 一部の機能が廃止されている
- デフォルト値が変わっている