

中小企業のクラウド活用ガイド

〈2026年版〉AWS・Azure・Google Cloudの選び方と安全・低コスト運用

— 3大クラウドを「同じ物差し」で比べ、失敗しない運用を始めるための実務書 —

- ✓ 3社を8つの観点で横比較した早見表
- ✓ 自社に合うクラウドを決める3つの判断軸
- ✓ 最初にやるべきセキュリティ設定と優先順位
- ✓ 請求額を膨らませないコスト管理の仕組みとチェックリスト20

発行：Harmonic Society株式会社

（千葉市 | ホームページ制作・AI導入支援・IT顧問）

はじめに — 「どのクラウドを選ぶか」 より大切なこと

「社内サーバーの保守期限が近い。クラウドに移すべきか」「AWSとAzureとGoogle Cloud、結局どれがいいのか」「クラウドにしたら請求額が読めなくなった」——クラウドに関するご相談は、この3つに集約されます。

クラウドの3大事業者であるAWS（Amazon Web Services）、Microsoft Azure、Google Cloudは、いずれも中小企業の業務システムやWebサービスを動かすのに十分な機能を備えています。実は、**事業者選**
びで大きく失敗することはあまりありません。失敗の多くは、選んだ後の「設計と運用」で起きます。管理者アカウントに多要素認証を設定していなかった、データベースをインターネットに公開したままだった、開発用のサーバーを止め忘れて月末に驚いた——どれも、どのクラウドでも起こりうる事故です。

そこで本書は、3社を**同じ観点（ストレージ・データベース・サーバーレス・ネットワーク・ID管理・監視・CI/CD・コスト管理）で横に並べ**、「名前は違うが、やるべきことは同じ」部分と「3社で考え方が違う」部分を整理しました。3社の用語の対応表としても使えるように作っています。

本書は、当社ブログのクラウド関連記事30本（AWS・Azure・Google Cloudの各サービス解説）を、導入検討から運用までの順序で一冊に再構成したものです。読み終えたとき、「自社はどのクラウドを選び、最初の1か月で何を設定すべきか」を説明できることをゴールにしています。

本書の情報について：サービス名・機能・料金体系・無料枠は**2026年9月時点**の情報に基づきます。クラウドの料金や無料枠、サービス名称は頻繁に変更されるため、本書では具体的な単価をあえて掲載せず、比較の軸と考え方を中心に解説しています。導入・契約の判断にあたっては、必ず各社の公式サイトで最新情報をご確認ください。文中の効果・割合は当社ブログ記事で紹介した一般的な目安であり、効果を保証するものではありません。

第1章 クラウドの基本 — 何を任せ、何が自社に残るのか

1-1. クラウドの3つの使い方

クラウドの使い方は、事業者にどこまで任せるかで大きく3つに分かれます。中小企業にとっての基本方針は、**任せられるところは任せる**ことです。

図表1：クラウドの利用形態と運用負担

形態	任せる範囲	3社の代表例	自社の運用負担
IaaS（仮想サーバー）	物理機器・ネットワーク	EC2 / Virtual Machines / Compute Engine	大（OS更新・監視も自社）
PaaS（マネージド）	上記+OS・ミドルウェア	RDS / App Service / Cloud SQL	中（設定と監視が中心）
サーバーレス	上記+サーバーの存在そのもの	Lambda / Functions / Cloud Run	小（コードと設定のみ）

仮想サーバー（IaaS）は自由度が高い反面、OSのセキュリティ更新やバックアップを自社で担います。データベースの運用記事でも繰り返し述べているとおり、**パッチ適用・バックアップ・障害時の切り替えをクラウド側に任せられるマネージドサービス**を選ぶだけで、少人数の情報システム担当者の負担は大きく下がります。

1-2. 「共有責任モデル」を最初に理解する

クラウドのセキュリティは、事業者と利用者の**共有責任モデル**で成り立っています。データセンターの物理的な安全や基盤の保守は事業者の責任ですが、「誰にどの権限を与えるか」「データを公開するか」「通信をどこまで許可するか」は、**どのクラウドでも利用者の責任**です。

- **事業者の責任**：建物・電源・物理サーバー・基盤ネットワーク・マネージドサービスの基盤
- **利用者の責任**：アカウントとパスワード、アクセス権限、ネットワーク設定、データの公開範囲、暗号化の選択、バックアップの復元確認

「クラウドにしたから安全」ではなく、「安全に使うための設定が必要な道具を手に入れた」と捉えることが、本書全体の前提です。

第1章 クラウドの基本（続き） — 3社の選び方

1-3. 自社に合うクラウドを決める3つの判断軸

3社の機能差は、中小企業の一般的な用途（Webサイト、業務システム、ファイル保管、データベース）ではほとんど決定打になりません。現実的な判断軸は次の3つです。

- 1. 既存のID基盤・業務ツールとの相性**：社員のアカウントをMicrosoft 365で管理しているならAzure（Microsoft Entra ID）との連携が自然です。Google Workspaceを使っているならGoogle Cloudとのアカウント統合がスムーズです。
- 2. 任せられる人・会社がいるか**：社内や取引先のエンジニアが経験のあるクラウドを選ぶことは、機能差以上に重要です。AWSは利用者が多く、日本語の情報や対応できる会社を見つけやすい傾向があります。
- 3. 作りたいものの性格**：Windows・.NET・SQL Server中心の既存システムならAzure、コンテナを手軽にサーバーレスで動かしたい・データ分析（BigQuery）を活用したいならGoogle Cloud、幅広いサービスから組み合わせたいならAWSが候補になります。

図表2：状況別・第一候補の目安

自社の状況	第一候補	理由
Microsoft 365中心・Windows系システムが多い	Azure	Entra IDでSSO・MFAを一元管理できる
Google Workspace中心・データ分析を重視	Google Cloud	アカウント統合、BigQuery・Cloud Runが手軽
外注先・採用候補の経験を重視、選択肢を広く	AWS	利用者と情報量が多く、サービスの幅が広い
まだ決め手がない	担当者が扱えるもの	運用できることが最大の安全策

1-4. 「全部1社」にこだわらない

メールや社員アカウントはMicrosoft 365、Webサイトや業務アプリはAWS、というように**用途ごとに使い分けるのは珍しくありません**。ただし、管理画面・請求・権限管理が分散すると運用負担は増えます。最初は「メインのクラウドを1つ決め、例外は理由がある場合だけ」とするのが無難です。

第2章 3社横比較 — 8つの観点の早見表

同じ役割のサービスでも、3社で名前がまったく違うことが比較を難しくしています。下の表は、本書で扱う8つの観点ごとに、中小企業がまず使う代表的なサービスを並べたものです（2026年9月時点）。

図表3：AWS・Azure・Google Cloud 主要サービス対応表

観点	AWS	Microsoft Azure	Google Cloud
ストレージ	Amazon S3	Blob Storage	Cloud Storage
データベース	RDS / Aurora	Azure SQL Database	Cloud SQL
サーバーレス	Lambda / ECS on Fargate	Azure Functions / App Service	Cloud Run / Cloud Run functions
ネットワーク	VPC（リージョン単位）	Virtual Network（リージョン単位）	VPC（グローバル）
ID・権限管理	IAM / IAM Identity Center	Microsoft Entra ID / Azure RBAC	Cloud IAM / サービスアカウント
監視	CloudWatch	Azure Monitor / Application Insights	Cloud Monitoring / Cloud Logging
CI/CD	CodePipeline / CodeBuild	Azure Pipelines（Azure DevOps）	Cloud Build / Artifact Registry
コスト管理	Cost Explorer / AWS Budgets	Cost Management / Azure Advisor	請求レポート / 予算アラート

この表の読み方

- ・ **「やるべきこと」は3社共通です。** ストレージの公開を防ぐ、データベースを外部から隔離する、管理者に多要素認証（MFA）を設定する、予算アラートを設定する——第3章以降の内容は、どのクラウドにもほぼそのまま当てはまります。
- ・ **違いが出るのは「管理の単位」と「考え方」です。** 次ページで4つの違いを整理します。
- ・ 「Cloud Run functions」は2024年に旧Cloud Functionsから改称されたサービスです。名称は変わりうるため、社内資料には役割（例：サーバーレス関数）も併記しましょう。CI/CDはGitHub Actionsなど外部ツールとの組み合わせも一般的です。

第2章 3社横比較（続き） — 設計思想の4つの違い

2-1. リソースをまとめる「単位」が違う

図表4：管理単位と権限・費用の区切り方

項目	AWS	Azure	Google Cloud
課金・分離の基本単位	アカウント	サブスクリプション	プロジェクト
その中のまとまり	タグで分類	リソースグループ	ラベルで分類
複数単位の束ね方	AWS Organizations	管理グループ	組織・フォルダ

本番・開発を**アカウント（AWS）／サブスクリプション（Azure）／プロジェクト（Google Cloud）単位で分けておく**と、権限と請求を分離でき、開発環境の設定ミスが本番に波及するリスクを下げられます。小規模なら最初は1つでも構いませんが、「本番と開発を同じ場所に混ぜない」ことは早めに決めておきましょう。

2-2. ネットワークの広さが違う

AWSのVPCとAzureのVirtual Networkは**リージョン（地域）ごと**に作ります。一方、Google CloudのVPCは**グローバル**で、1つのVPCの中に東京や大阪などリージョン単位のサブネットを置く構造です。複数拠点に展開するときの設計が変わりますが、国内1リージョンで始める中小企業では大きな差にはなりません。

2-3. ID管理の出発点が違う

AzureはMicrosoft Entra ID（旧Azure Active Directory）という**社員アカウントの基盤そのもの**がクラウドの入口になっており、Microsoft 365と同じアカウントでSSO（シングルサインオン）やMFAを管理できます。AWSはIAMとIAM Identity Center、Google CloudはGoogleアカウントとCloud IAMで権限を管理します。**既存の社員アカウント基盤と揃えられるか**が、日々の運用のしやすさを左右します。

2-4. 割引の仕組みが違う

3社とも「1年・3年の利用を約束すると割安になる」仕組み（AWSのSavings Plans・リザーブドインスタンス、Azureの予約・Savings Plan、Google Cloudの確約利用割引）を持っています。加えてGoogle Cloudには、一部の仮想マシンを月の中で長く使うと**申し込み不要で自動的に割り引かれる継続利用割引**があります。詳しくは第6章で扱います。

第3章 データとアプリの置き場所を選ぶ

3-1. ストレージ — ファイル保管は「オブジェクトストレージ」が基本

S3、Blob Storage、Cloud Storageはいずれも**オブジェクトストレージ**（ファイルを「データ＋付帯情報」として保存し、URLでアクセスする方式）です。3社とも99.999999999%（イレブンナイン）の耐久性を掲げ、容量の事前確保が不要で、使った分だけ課金されます。社内NASやファイルサーバーのディスク故障リスクから解放されるのが最大の利点です。

コストの鍵は「アクセス頻度に応じた保存クラスの使い分け」です。3社とも、頻繁に使うデータ用の標準クラスから、ほとんど取り出さない長期保存用のクラスまで段階があり、保存料が安いクラスほど取り出し料や最低保存期間が発生します。

図表5：アクセス頻度別の保存クラス（名称の対応）

アクセス頻度の目安	AWS (S3)	Azure (Blob)	Google Cloud
頻繁に使う	Standard	ホット層	Standard
月1回程度	Standard-IA	クール層	Nearline
四半期に1回程度	Glacier Instant Retrieval	コールド層	Coldline
年1回未満・法定保存	Glacier Deep Archive など	アーカイブ層	Archive

「作成から30日後に1段下のクラスへ、1年後にアーカイブへ、7年後に削除」のように、**ライフサイクルポリシー（経過日数による自動移行・自動削除）**を設定しておけば、手作業なしで保存コストを下げられます。アクセスパターンが読めないデータには、S3 Intelligent-Tieringのような自動階層化も選択肢です。

3-2. ストレージで事故を起こさないための共通ルール

- **公開を防ぐ設定を最初に有効化**：S3のパブリックアクセスブロック、Azureのストレージアカウント単位のパブリックアクセス無効化、Google Cloudのパブリックアクセス防止
- **一時的な共有は期限付きURLで**：署名付きURL（AWS・Google Cloud）やSASトークン（Azure）を、最短の期限・最小の権限で発行する
- **誤削除に備えてバージョニングを有効化**：上書き・削除しても以前の版に戻せる
- **暗号化は既定で有効**：3社とも保存データは標準で暗号化される。監査要件があれば自社管理の鍵（KMS等）を検討

第3章 データとアプリの置き場所を選ぶ（続き）

3-3. データベース — 迷ったら「マネージドDB」

業務システムの心臓部であるデータベースは、仮想サーバーに自分でインストールするより、**マネージドデータベース**（構築・パッチ適用・バックアップを事業者が代行するサービス）を選ぶのが基本です。

図表6：3社のマネージドデータベース比較

項目	AWS RDS / Aurora	Azure SQL Database	Google Cloud SQL
主なエンジン	MySQL・PostgreSQL・SQL Server ほか	SQL Server系（MySQL等は別サービス）	MySQL・PostgreSQL・SQL Server
自動バックアップ	保持1～35日、任意時点へ復元	保持1～35日、長期保持も可	毎日自動、任意時点へ復元
高可用性	Multi-AZ構成	サービス階層に組み込み	リージョナル構成（ゾーン間切替）
向いているケース	選択肢の広さ、Aurora活用	既存SQL Server・.NET資産	Cloud Runとの組み合わせ

押さえるべき共通の設定は次のとおりです。

- ・ **インターネットから直接つながらない**：プライベートなネットワーク（第4章）に配置し、アプリからのみ接続させる
- ・ **バックアップ保持期間を延ばす**：初期値のままにせず、業務に合わせて延長（ブログでは14日以上を推奨）。リリース前には手動バックアップ（スナップショット）を取る
- ・ **復元テストをする**：「取れているはず」ではなく、定期的に実際に戻せるか確認する
- ・ **削除保護と暗号化を有効化**：誤操作によるデータベース削除を防ぐ
- ・ **本番は冗長構成に**：障害時に別のゾーン（同じ地域内の別データセンター群）へ自動で切り替わる構成を選ぶ

3-4. 小さく始められる選択肢

アクセスが少ない・不規則なシステムには、使わない時間はコンピュータ料金がからない**サーバーレス型のデータベース**（Aurora Serverless v2、Azure SQL Databaseのサーバーレス層など）も候補です。一方、常時アクセスがある本番システムでは通常のインスタンス＋長期割引の方が割安な場合もあります。開発環境は小さい構成から始め、性能監視（第5章）の結果を見てサイズを調整しましょう。

第3章 データとアプリの置き場所を選ぶ（続き）

3-5. サーバーレス — 「サーバーを持たない」という選択

サーバーレスとは、サーバーの構築・OS更新・台数調整を事業者任せ、**リクエストやイベントがあったときだけ処理が動き、その分だけ課金される**仕組みです。中小企業にとっては「専任のインフラ担当がいなくても本番運用できる」ことが最大の価値です。

図表7：用途別・3社のサーバーレス/コンテナ実行環境

用途	AWS	Azure	Google Cloud
小さな自動処理（定期実行など）	Lambda	Azure Functions	Cloud Run functions
Webアプリ・API	ECS on Fargate / Lambda	App Service	Cloud Run
コンテナをサーバー管理なしで	ECS on Fargate	Container Apps	Cloud Run

選び方の目安：毎朝の集計、アップロード画像の縮小、Webhookの受け口といった**定型業務の自動化**は関数型が手軽です。**Webアプリを載せたい**なら、コードを置くだけで動くApp Service、コンテナをそのまま動かせるCloud Run・ECS on Fargateが候補です。Kubernetesのような複雑な仕組みを使わずに済むことが、中小企業向きの理由です。

3-6. サーバーレスの注意点

- **コールドスタート**：しばらく使われないと、最初の応答に数秒かかることがあります。速度が重要なら最小インスタンス数を1以上に（その分の費用は発生）。
- **上限の設定**：アクセス急増で費用が膨らまないよう、**最大インスタンス数などの上限を必ず設定**します。
- **実行時間の制限**：関数型には1回あたりの実行時間の上限があります。長時間の処理はコンテナ型に分けます。
- **重複実行への備え**：同じイベントが重複して届くことがあるため、何度実行しても結果が変わらない作り（冪等性）にします。
- **無料枠**：3社とも従量課金のサーバーレスに毎月の無料枠があり（2026年9月時点）、小規模な自動化なら費用がほとんどかからない場合もあります。内容は変更されるため公式情報で確認を。

第4章 安全の土台 — ネットワーク設計

4-1. 基本形は「公開する層」と「隠す層」の分離

クラウド上のネットワーク（VPC/Virtual Network）は、社内LANをクラウドに作るようなものです。3社とも基本形は同じで、**外から受け付ける層と、直接触らせない層を分けます。**

- **公開する層（パブリックサブネット）**：ロードバランサー（アクセスを振り分ける装置）など、入口だけを置く
- **隠す層（プライベートサブネット）**：アプリとデータベースを置き、入口からの通信だけ許可
- **外への通信**：プライベートな層からはNATゲートウェイ（Google CloudはCloud NAT）経由で出る

4-2. 3社のファイアウォール機能の違い

図表8：通信を制御する仕組みの比較

項目	AWS	Azure	Google Cloud
主な仕組み	セキュリティグループ+ネットワークACL	NSG	VPCファイアウォールルール
拒否ルール	セキュリティグループは許可のみ（拒否はACL）	許可・拒否を優先度順に評価	許可・拒否を優先度順に評価
対象の指定	セキュリティグループ同士の参照	サービスタグ等	ネットワークタグ・サービスアカウント
注意点	開発時の全開放を本番に持ち込まない	全サブネットにNSGを関連付ける	既定VPCは許可が緩いため独自VPCを作る

共通の原則は「既定で拒否し、必要な通信だけ許可」です。管理用のSSH（22番）やリモートデスクトップ（3389番）をインターネットに開けたままにするのは典型的な失敗です。管理アクセスには、Azure Bastionのような「管理ポートを公開せずに接続する仕組み」を使いましょう。

4-3. クラウドのサービスへは「裏道」でつなぐ

ストレージやデータベースなどクラウドのサービスへ、インターネットを経由せず事業者のネットワーク内で接続する仕組みがあります（AWSのVPCエンドポイント、Azureのプライベートエンドポイント、Google CloudのプライベートGoogleアクセスなど）。**安全性が上がり、NAT経由の通信料も減らせる**場合があります。AWSのS3向けゲートウェイ型エンドポイントは追加料金なしで使えます。

第4章 安全の土台（続き） — ネットワーク設計の失敗パターン

ネットワークは後から作り直すのが最も難しい部分です。ブログで取り上げた「よくある失敗」を、3社共通の形でまとめます。

図表9：ネットワーク設計の失敗パターンと対策

失敗パターン	何が起きるか	対策
アドレス範囲を小さく取りすぎる	後からIPアドレスが足りなくなる	余裕をもって確保（/16が一般的）
他ネットワークと範囲が重複	拠点VPNや別VPCとつなげない	社内LAN・他環境と重ならない範囲を台帳管理
開発時の「全開放」を本番へ	攻撃の入口になる	本番投入前にルールを棚卸し
単一ゾーン構成	データセンター障害で全停止	本番は2ゾーン以上に分散
通信ログを取っていない	障害・不正アクセスの原因が追えない	フローログを作成直後に有効化
既定のネットワークをそのまま本番利用	不要な許可や範囲が残る	本番用に独自のネットワークを作る

4-4. 設定は「コード」で残す

画面操作だけで構成を作ると、誰が何を変えたか分からず、同じ環境の再現も困難です。3社の記事で共通して推奨しているのが**laC（Infrastructure as Code：構成をコードで記述・管理する手法）**です。3社すべてに対応するTerraformのほか、AWS CloudFormation、AzureのBicepなど各社純正の仕組みもあります。本番と同じ構成を検証環境にすぐ再現でき、変更を承認の対象にできます。最初は**ネットワークと権限の設定から**コード化するのが効果的です。

4-5. 組織としての「ガードレール」

人数が増えてきたら、個々の設定ミスを組織として防ぐ仕組みを検討します。Azure Policy、Google Cloudの組織ポリシー、AWS Organizationsのサービスコントロールポリシーを使うと、「許可していない地域にリソースを作れない」「必須タグのないリソースを作れない」といったルールを強制できます。危険な設定の自動検出には、Microsoft Defender for Cloud、Google CloudのSecurity Command Center、AWS Trusted Advisorなどが使えます。

第4章 安全の土台（続き） — ID・権限管理

4-6. 最も重要なのは「最強の管理者アカウント」の保護

クラウドのセキュリティ事故の多くは、高度な攻撃ではなく**アクセス管理の不備**——過剰な権限、漏えいしたアクセスキー、削除し忘れた退職者のアカウント——から起きます。IAM（Identity and Access Management：誰が何をできるかを管理する仕組み）の設計は、3社共通で最優先の仕事です。

図表10：3社のID・権限管理の対応

やること	AWS	Azure	Google Cloud
最上位アカウントの保護	ルートユーザーにMFA、日常利用しない	グローバル管理者は最小人数・MFA必須	オーナーは最小人数・MFA必須
人の権限管理	IAM Identity Center+グループ	Entra ID+Azure RBAC	Googleグループ+IAMロール
プログラムの権限	IAMロール	マネージドID	サービスアカウント
不要権限の発見	IAM Access Analyzer	アクセスレビュー（上位ライセンス）	IAM Recommender
操作の記録	CloudTrail	アクティビティログ	Cloud Audit Logs

4-7. 3社共通の5原則

- 全員にMFA（多要素認証）**：パスワードが漏れても不正ログインを防ぐ最も効果的な対策です。
Microsoftは、MFAでアカウント侵害の99.9%を防げるとしています。管理者から始め、段階的に全社員へ広がります。
- 最小権限**：業務に必要な権限だけを付与します。AWSのAdministratorAccess、Google Cloudの基本ロール（オーナー／編集者）のような広い権限を「とりあえず」付与しないこと。
- 個人ではなくグループに権限を付ける**：人の入れ替わりがあっても、グループの出し入れだけで済みます。
- パスワードや鍵をコードに書かない**：プログラムにはIAMロール／マネージドID／サービスアカウントを使い、長期有効なアクセスキーは極力作らない。GitHub Actionsなどからの接続も、鍵を使わない連携方式（OIDC）が3社で使えます。
- 定期的な棚卸し**：90日以上使われていないアカウントやキー、退職者・異動者の権限を定期的に削除します。

第5章 運用を自動化する — 監視

5-1. 「ユーザーからの電話で障害に気づく」をなくす

「サーバーが落ちていることに、お客様からの連絡で初めて気づいた」——監視がない環境では珍しくありません。3社とも主要サービスの基本的な数値（メトリクス）は**追加設定なしで自動収集**されています。やるべきことは、**何を見て、いつ誰に知らせるか**を決めることです。

図表11：3社の監視サービスの対応

機能	AWS	Azure	Google Cloud
数値の監視・アラート	CloudWatch（メトリクス・アラーム）	Azure Monitor（メトリクス・アラート）	Cloud Monitoring
ログの集約・検索	CloudWatch Logs / Logs Insights	Log Analytics（KQLで検索）	Cloud Logging
アプリの性能監視	CloudWatch Application Signals 等	Application Insights	Cloud Trace 等
外形監視（死活監視）	CloudWatch Synthetics	可用性テスト	アップタイムチェック

仮想サーバーのメモリ・ディスク使用率は、3社とも**エージェント（監視用ソフト）の導入**が必要です。

5-2. 最低限の監視セット

- ・ **外形監視**：外部から定期アクセスし、応答がなければ通知（複数地点で失敗したときだけ通知すると誤報が減る）
- ・ **エラー数**：Webアプリの5xxエラー（サーバー側のエラー）の増加
- ・ **リソース使用率**：CPU・メモリ・ディスク（ディスクは85%前後で警告が目安）
- ・ **データベースとサーバーレス**：CPU・接続数・空き容量、関数のエラー数と実行時間

5-3. アラート疲れを防ぐ

通知が多すぎると、重要な通知まで無視されます。**重要度で通知先を分け**（緊急は電話・SMS、警告はチャット、情報はメール）、「5分間×3回連続で超えたら」のように**継続時間を条件にして**一瞬の変動では鳴らさないこと。閾値は最初は緩めにし、運用しながら調整します。

第5章 運用を自動化する（続き） — ログ費用とCI/CD

5-4. 監視そのものの費用にも注意

監視は多くが従量課金で、特に**ログの量と保存期間**が費用を押し上げます（本番で詳細な開発用ログを出し続け、月数万円の追加費用が出る例も）。本番のログは原則INFO以上にし、保存期間を用途別に決め、長期保存分は安価なストレージへ書き出しましょう。

5-5. CI/CD — 手作業のリリースをやめる

CI/CD（継続的インテグレーション／継続的デリバリー）とは、プログラムの変更を登録するたびに**テスト・組み立て・本番反映を自動で行う仕組み**です。手順漏れやテストのやり忘れをなくし、履歴も残ります。

図表12：3社のCI/CDサービスと安全な反映方法

項目	AWS	Azure	Google Cloud
自動化の中心	CodePipeline+CodeBuild	Azure Pipelines（YAMLで定義）	Cloud Build（cloudbuild.yaml）
安全な切り替え	CodeDeployのBlue/Green・Canary	App Serviceのデプロイスロット	Cloud Runのトラフィック分割
本番前の承認	手動承認ステージ	環境ごとの承認ゲート	承認フローの設定
機密情報の保管	Secrets Manager / Parameter Store	Key Vault	Secret Manager

Blue/Greenは新旧2環境を用意して切り替える方式、**Canary**は一部の利用者から新版に切り替えて確かめる方式で、どちらも問題があればすぐ戻せます。

5-6. 小さく始めて段階的に自動化する

- まずは「変更を登録したら自動でテストが走る」だけを作る
- 検証環境への自動反映を加える
- 本番反映は**手動承認を挟んで**自動化し、テストの充実に合わせて承認を減らす

パスワードやAPIキーは設定ファイルに書かず上表のサービスから参照し、CI/CDには**専用の最小権限**を与えます。

第6章 コスト管理 — 請求額に驚かないために

6-1. クラウドの費用は「4つの要素」でできている

「クラウドは安い」と言われる一方で、「気づいたら請求額が想定を大きく超えていた」という経験も多く聞かれます。従量課金は、**管理しなければ際限なく増える**仕組みでもあります。3社とも、費用は主に次の4つで構成されます。

- **計算（コンピューティング）**：仮想サーバー・コンテナ・関数の稼働時間やリソース量。多くの企業で最大の割合
- **保存（ストレージ）**：データ量と保存期間、保存クラス
- **通信（ネットワーク）**：主にクラウドから外部への送信データ。受信は無料のことが多い
- **リクエスト・処理**：API呼び出し回数、データ分析の処理量など

見落としがちなのは、**止めたつもりでも課金が続くもの**です。停止した仮想サーバーのディスク、どこにも関連付けていない固定IPアドレス、削除したサーバーの残骸のディスクやスナップショット、使っていない環境のNATゲートウェイやロードバランサーなどが典型例です。

6-2. 可視化 — 「何にいくら」を分かるようにする

図表13：3社のコスト管理機能の対応

機能	AWS	Azure	Google Cloud
費用の分析	Cost Explorer	Cost Management（コスト分析）	請求レポート／BigQueryへの書き出し
予算と通知	AWS Budgets	予算（Cost Management）	予算とアラート
削減の提案	Trusted Advisor 等	Azure Advisor	Recommender
費用の分類	コスト配分タグ	タグ	ラベル

タグ（ラベル）付けは最初に決めるルールです。「環境（本番／検証／開発）」「プロジェクト名」「担当部署・責任者」の3～4種類を全リソースに付けると、「開発環境が本番と同じくらい費用を使っていた」といった発見が可能になります。後からまとめて付けるのは大変なので、Azure Policyなどで「タグのないリソースは作れない」ようにする方法もあります。

第6章 コスト管理（続き） — 予算アラートと削減の打ち手

6-3. 予算アラートは「多段階」＋「予測」で

3社とも、月の予算を設定して一定割合で通知する機能があります。**50%・80%・100%の実績通知に、「このペースだと月末に超えそう」という予測ベースの通知**を組み合わせると、月末を待たずに手を打てます。

注意：予算アラートは基本的に「通知」であり、予算を超えても課金は自動で止まりません。自動停止まで行わず、通知をきっかけに開発環境を止める処理（Azureのアクショングループ、Google CloudのPub/Sub連携など）を別途組みます。

6-4. 削減の打ち手 — 即効性の高い順に

- 未使用リソースの削除：**3社の提案機能（Trusted Advisor／Azure Advisor／Recommender）が、低使用率のサーバーや未使用のディスク・IPアドレスを教えてください。
- 開発・検証環境の夜間・休日の自動停止：**平日の業務時間だけ動かせば稼働時間は大幅に減ります。
- サイズの適正化：**CPU使用率が常に低いサーバーは1サイズ下げる。
- 保存クラスの自動移行とCDN**（CloudFront／Azure Front Door／Cloud CDN）による通信量の削減。
- 長期割引の適用：**3か月以上安定稼働の本番リソースから、対象の柔軟な方式（Compute Savings Plans等）を控えめな金額で。

図表14：料金モデルの使い分け（3社共通の考え方）

料金モデル	3社の名称	向いている用途
従量課金（オンデマンド）	各社標準	検証中・負荷が読めないもの
利用確約型の割引（1年・3年）	Savings Plans・RI／予約・Savings Plan／確約利用割引	常時動く本番（使わなくなっても支払いは続く）
余剰枠の格安利用（中断あり）	スポットインスタンス／Spot VM	バッチ処理・検証（本番のWeb・DBには不向き）
自動割引	Google Cloudの継続利用割引（一部のVM）	申し込み不要

第7章 続けるための仕組み — 優先順位とロードマップ

7-1. 何から手を付けるか — 対策の優先順位

やるべきことが多すぎて手が止まるのが、中小企業で最もよくある状況です。設計レビューの記事で紹介した優先順位の考え方を、3社共通の形に整理しました。

図表15：対策の優先順位と対応目安

優先度	対応すべき項目の例	対応目安
最優先	管理者のMFA未設定、ストレージやDBの外部公開、暗号化なし	即座に
高	単一ゾーン構成、バックアップ未設定・復元未確認	1か月以内
中	監視・アラート不足、手作業の構築・デプロイ	3か月以内
低	コスト最適化、性能改善の余地	次の見直しまで

7-2. 導入・移行の3段階ロードマップ

図表16：クラウド活用の段階的な進め方

段階	期間の目安	主な取り組み
1. 土台づくり	1～2か月	管理者MFA・予算アラート・操作ログ、本番と開発の分離
2. 運用の自動化	3～6か月	監視、バックアップと復元テスト、IaC・CI/CD、手順の文書化
3. 最適化	6か月以降	コスト最適化、長期割引、性能改善、定期レビュー

移行の最初の一步は、**静的なWebサイトやファイルのバックアップ先など、止まっても業務影響の小さいもの**から。手順と運用に慣れてから基幹システムへ広げるのが現実的です。

7-3. 定期的な「設計レビュー」を習慣にする

AWSとAzureは、良い設計を体系化した**Well-Architected Framework**（AWSは運用・セキュリティ・信頼性・パフォーマンス効率・コスト最適化・サステナビリティの6本柱、Azureはサステナビリティを除く5本柱）と、質問に答えるだけで現状を評価できる無料ツールを提供しています。Google Cloudも同種の設計指針を公開しています。完璧を目指さず、**半年～四半期に1回**、または大きな機能追加や障害の後に見直しましょう。

巻末付録1 クラウド運用チェックリスト20

【アカウントと権限】

- ☐ 1. 最上位の管理者アカウントにMFAを設定し、日常作業では使っていない
- ☐ 2. 全ユーザーにMFAを必須にしている
- ☐ 3. 権限は個人ではなくグループ単位で付与し、最小権限にしている
- ☐ 4. プログラムには長期のアクセスキーではなくロール等を使っている
- ☐ 5. 退職者・異動者の権限と、90日以上未使用のアカウント・キーを定期的に削除している
- ☐ 6. 操作ログ（CloudTrail／アクティビティログ／Cloud Audit Logs）を記録・保管している

【ネットワークとデータ】

- ☐ 7. データベースとアプリはプライベートな層に置き、インターネットから直接つながらない
- ☐ 8. SSH・リモートデスクトップの管理ポートをインターネットに公開していない
- ☐ 9. ストレージの公開防止設定を有効にしている
- ☐ 10. 本番はネットワーク・アカウント等の単位で開発環境と分離している
- ☐ 11. 自動バックアップの保持期間を業務に合わせて設定し、復元テストを実施した
- ☐ 12. 本番のデータベースとアプリは複数ゾーンに分散している

【運用の自動化】

- ☐ 13. 外形監視とエラー数のアラートを設定し、通知先を重要度で分けている
- ☐ 14. ログの保存期間を用途別に決めている
- ☐ 15. 本番反映はCI/CDで自動化し、機密情報は専用サービスで管理している
- ☐ 16. ネットワークと権限の構成をIaC（Terraform等）で管理している

【コスト管理】

- ☐ 17. 予算アラートを多段階＋予測ベースで設定している
- ☐ 18. 全リソースに環境・プロジェクト・責任者のタグ（ラベル）を付けている
- ☐ 19. 開発・検証環境に夜間・休日の自動停止を設定している
- ☐ 20. 月1回、コストの前月比と未使用リソースを確認している

判定の目安：10個以下なら、まず1・2・7・9・17（事故と高額請求を防ぐ5項目）から。16個以上なら定期レビューと長期割引の検討へ。

巻末付録2 3社の用語・サービス名対応表

同じ役割の機能を3社がどう呼んでいるかの対応表です（2026年9月時点。名称は変更される場合があります）。

図表17：用語・サービス名の対応表

役割	AWS	Microsoft Azure	Google Cloud
仮想サーバー	EC2	Virtual Machines	Compute Engine
Webアプリの実行基盤	ECS on Fargate 等	App Service	Cloud Run
サーバーレス関数	Lambda	Azure Functions	Cloud Run functions
オブジェクトストレージ	S3	Blob Storage	Cloud Storage
マネージドRDB	RDS／Aurora	Azure SQL Database 等	Cloud SQL
仮想ネットワーク	VPC	Virtual Network（VNet）	VPC
ファイアウォール	セキュリティグループ	NSG	ファイアウォールルール
CDN	CloudFront	Azure Front Door	Cloud CDN
ID・権限管理	IAM／IAM Identity Center	Microsoft Entra ID／Azure RBAC	Cloud IAM
機密情報の保管	Secrets Manager	Key Vault	Secret Manager
監視・ログ	CloudWatch	Azure Monitor／Log Analytics	Cloud Monitoring／Cloud Logging
操作の監査ログ	CloudTrail	アクティビティログ	Cloud Audit Logs
CI/CD	CodePipeline／CodeBuild	Azure Pipelines	Cloud Build
コスト分析	Cost Explorer	Cost Management	請求レポート

おわりに — 発行元・ご相談窓口

クラウドは、少人数の会社でも大企業と同じ品質の基盤を、使った分だけの費用で使える道具です。一方で、設定を誤れば情報漏えいや想定外の請求につながります。本書でお伝えしたかったのは、**どのクラウドでも「やるべきことはほぼ同じ」であり、最初の設定と、続ける仕組みで結果が決まる**ということです。まずは巻末のチェックリストで現状を確認するところから始めてみてください。

Harmonic Society株式会社

千葉市を拠点に、中小企業のホームページ制作・AI導入支援・IT顧問を提供しています。当社の自社サイトもAWS（S3・CloudFront・Lambda など）で運用しており、本書のもとになったクラウド解説記事群は、日々の構築・運用の実務を踏まえてまとめたものです。オンライン対応・全国OK。

サービスのご案内

サービス	内容・価格
IT顧問ライトプラン	クラウド・AI・ITの継続相談を月1.5万円から
ホームページ制作	月額制 月5,500円～（初期0円）／LP制作 1本2.5万円～（いずれも税込）
AI導入支援	要件整理から導入・定着まで伴走

まずは30分の無料相談から

「自社にはどのクラウドが合うか」「今の設定で危ないところはないか」「請求額を下げられないか」——本書を読んで浮かんだ疑問を、そのままお持ちください。売り込みはしません。

- Web：<https://harmonic-society.co.jp/contact/>
- お役立ち資料一覧：<https://harmonic-society.co.jp/whitepaper/>

本書の内容は2026年9月時点の情報に基づきます。サービス名・機能・料金体系・無料枠は変更される場合があります。掲載した効果・割合は一般的な目安であり、効果を保証するものではありません。AWS、Microsoft Azure、Google Cloudおよび各サービス名は、各社の商標または登録商標です。無断転載を禁じます。