

インフラ自動化の標準化

Infrastructure as Codeによって任意のインフラをプロビジョニングおよび管理

インフラのプロビジョニングがクラウドと共に進化

多くの組織はクラウドの採用により、静的なインフラから動的なインフラへの移行が進みました。そのことにより、無限の拡張と配信、一時性と不変性の導入、複数の環境へのデプロイが可能になりました。

静的



- ・専用サーバー
- ・固定のリソース
- ・同一環境

動的

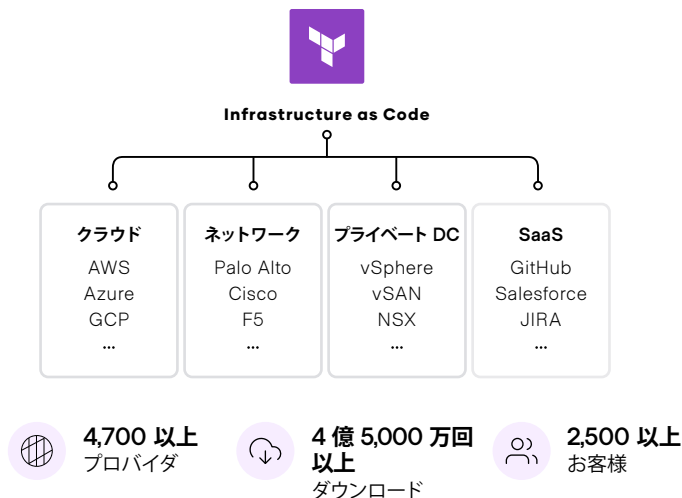


- ・オンデマンドでの容量提供
- ・無限のリソース
- ・混合環境

HashiCorp Terraform

あらゆるインフラをプロビジョニングおよび管理するインフラ自動化のスタンダード。

- ・ **Infrastructure as Code:** インフラをコードで自動化
- ・ **再利用可能なモジュールとレジストリ:** チーム利用で効率化
- ・ **ポリシーおよび自動適用:** Sentinel、OPA、Run Task を使用
- ・ **ヘルスチェック:** ドリフト検出、継続的な検証を実施
- ・ **セルフサービスのインフラ:** ノーコードプロビジョニングにより、Terraform に詳しくなくても、コードの記述なしでリソースの検出とデプロイが可能
- ・ **一時的なワークスペース:** 一時的なリソースを自動で破棄



メリット

5 倍 市場投入までの期間の短縮

共同作業に適した Infrastructure as Code でチームの生産性が向上

IT チームはプロビジョニングワークフローで Infrastructure as Code を作成、パブリッシュ、更新、削除することで、アプリケーションを迅速に配信できます。

25% コストの削減

クラウドのコンプライアンス確保および管理強化によってリスクとクラウドのコストを削減

単一のコントロールプレーンでマルチクラウドを可視化して管理することにより、シャドー IT やアイドリング、リソースのオーバープロビジョニングを回避できます。

96% リスクの軽減

ガードレールおよび再利用可能なモジュールを作成して適用

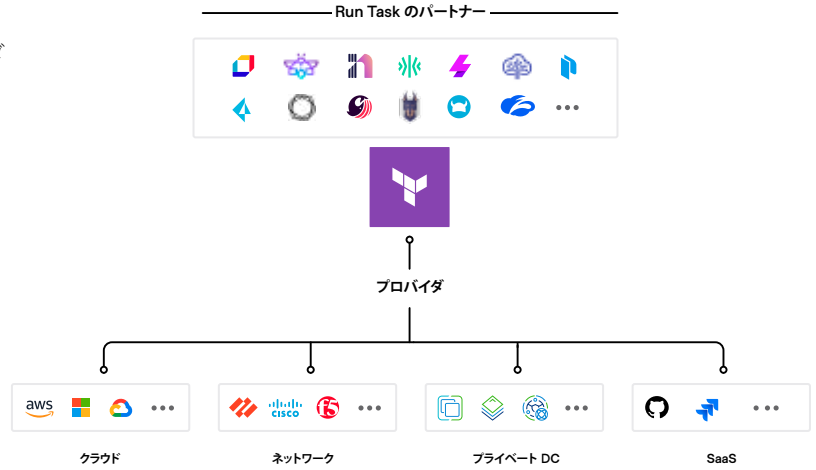
セキュリティ、コンプライアンス、運用上のベストプラクティスを踏まえ、承認済みのモジュールをパブリッシュしてポリシーを自動で適用します。

エコシステム

インフラのすべてと連携する API、ネイティブ統合、プロバイダプラグインの幅広いエコシステムを利用できます。

- ・ 4,700 以上のプロバイダ
- ・ 265 社以上のパートナーとの連携
- ・ 18,500 以上のモジュールで簡単に利用開始
- ・ 25 以上の Run Task でサードパーティに対してコストとセキュリティポリシーのチェックを実施
- ・ 80 以上のシステムインテグレーターを通じて、サービスを配信して迅速にオンボーディング

エコシステムの全容については、registry.terraform.io をご覧ください。



機能の比較

		Terraform Community	HCP Terraform クラウドホスト型			Terraform Enterprise セルフマネージド
			Standard	Plus	Premium	
導入 Infrastructure as Code により 自動化	Infrastructure as code HashiCorp Configuration Language (HCL) を使用してインフラを定義	✓	✓	✓	✓	✓
	Team collaboration ロールベースのアクセス制御、VCS への 接続、リモート運用を実現		✓	✓	✓	✓
導入 セキュリティおよび 管理のための 制御を構築	作成、テスト、パブリッシュ モジュールを作成し、連携済みテスト ワークフローによる検証を経てプライベート レジストリでモジュールをパブリッシュ		✓	✓	✓	✓
	ポリシーおよび適用 Terraform の実行中、インフラプロビジョ ニングの前にポリシーガードレールを適用		✓ * (利用制限あり)	✓	✓	✓
	ドリフト検出 実際のインフラを予期された状態と照合し、 変更点を継続的にチェック			✓	✓	✓
	モジュールの破棄 不慣れなユーザーが古い、あるいは孤立した モジュールバージョンを使うことを回避				✓	
拡張 セルフサービスの インフラでチーム、 開発者、BU をオ ンボーディング	ノーコードプロビジョニング エンドユーザーはコードを 1 行も記述することなく、 インフラを検出およびプロビジョニング可能			✓	✓	✓
	継続的な検証 運用直後からインフラの健全性を継続的に 検証			✓	✓	✓
	一時的なワークスペース 一時的なリソースと非アクティブな ワークスペースを自動で破棄			✓	✓	✓
	プライベート VCS へのアクセス、ポリシーの適用、 Run Task Terraform を管理する手間なく、高水準な セキュリティおよびガバナンス要件に対応				✓	