



GitHub

お客様事例

# お客様事例 | 世界的な成功の扉を開く

世界最大の開発者コラボレーションプラットフォームである GitHub は、HashiCorp ソリューションを通じて内部プロセスを補強し、ミッションクリティカルな機能をすばやく低価格で提供しています。

// インフラにより実現するイノベーション

# GitHub 社について

GitHub 社は開発者企業です。6,500 万人を超える世界中の開発者たちが GitHub を拠点にして、最善のコードを作成、共有し、市場に送り出しています。協同作業と問題解決を円滑に行える GitHub では、世界的に重要度の高いテクノロジーを開発するのも難しくはありません。

## 基本情報

---



6,500 万人以上の開発者



ロードバランシングに  
かかる時間を 96% 削減



ロードバランシングの  
再構成にかかる時間を  
30 分から 1 分に短縮



数千台ものノードに 400 以上  
の個別アプリケーション



オンボーディングにかかる  
コストと時間を削減



セキュリティ体制の  
大幅な改善

” Terraform を利用して、開発チーム向けのセルフサービス型ビジネスモデルを構築しました。一元的なゲートキーパーチームによる確認と承認のプロセスがなくなり、開発者間の摩擦が減りました”

SCOTT SANDERS 氏  
インフラ担当 VP

## ビジネスプロセスの欠陥を取り除く

GitHub は 6,500 万以上のユーザーを抱える世界最大の先進的な開発者コラボレーションプラットフォームです。アプリケーション開発や、異種クラウド環境をまたがるマイクロサービスの調整を支援し、それらの接続が安全に保護されるインフラを提供することを大きなビジョンに掲げています。

自社製のシークレット管理システムは手作業によるカスタム構成が必要で、数百ものアプリケーションやサービスを接続するためのコードも必要でした。さらに、キーとシークレットの管理も手作業でした。自社のサービスやアプリケーションを保護している静的なシークレットを変更すると、それに結び付けられているすべてのシステムでシークレットの変更が必要になります。

サーバーのロードバランシングなど、他の主要業務も手作業の割合が高く、時間のかかるものでした。チームはその時点で存在する各ホストに対し静的な構成を書き出すソフトウェアを作成しなければなりません。サービス容量を増やしたり、新しいサービスを追加したりする際は、特定のロードバランサに新しい構成が必要になることもあり、これが社内の全チームに悪影響を与えることもありました。

“シークレット管理からロードバランシングまで、あらゆる業務に自社製システムを使用していました。そのため、数百ものサービスと数千ものノードにわたって、膨大な数の統合とコーディングがそれぞれに必要でした” と、GitHub 社のインフラ担当バイスプレジデント Scott Sanders 氏は述べます。“しかし、アプリケーションやノード、ユーザーの数が増えるにつれ、このモデルの継続は困難になっていきます。ダイナミックに拡大するユーザー基盤を支えるため、より効率的で標準的な自動化が必要でした”

## 課題



バックエンドのインフラを拡張し、ユーザー基盤の拡大ニーズに応える



時間がかかる手作業のインフラプロビジョニングやサービスマッチング、  
シークレット管理を減らす



開発者が必要に応じて自分自身でインフラを構築できるようにする

## 設定したら後はお任せ

プラットフォームの成長に合わせて戦略的なインフラ投資を続けてきた GitHub 社でしたが、それまでのテクノロジスタックをより柔軟で俊敏な別のテクノロジに進化させる機会が訪れました。それは、ファーストクラスの開発者エクスペリエンスの提供に焦点を置き、複数のクラウドとプラットフォームに対応する特定の環境に依存しないソリューションを実現することへの転換でした。

GitHub 社が採用したのは HashiCorp の Terraform、Consul、Vault です。最善の組み合わせのオープンソースツールを稼働させてバックエンドのオペレーションを拡張し、社内のエンジニアリングチームや GitHub 製品の利用客からの増大するニーズに応えようと考えたためです。同社にとって案ずるべきは、製品をいかに効果的に拡張するかであって、業界内で解決済みの問題には時間をかけたくありませんでした。

同社が特に必要としていたのは、さまざまなオートメーション機能を備え、Azure や AWS、Google Cloud のワークロードとシームレスに連携できる実績のあるツールです。

“インフラ、ディスカバリ、シークレットを管理するためのツールが必要なことは明らかでした。基本的に自動で稼働し、堅牢なオートメーション機能を備え、設定と構成が簡単なツールです” と、Sanders 氏は説明します。“HashiCorp のソリューションは直感的で使いやすく、初期設定が完了すれば自動で機能してくれます。ですから私たちは、より価値の高い戦略や業務に専念できるのです”

同社のインフラソフトウェアエンジニアのひとり Aaron Brown 氏によると、GitHub のエンジニアは Terraform モジュールを利用し、開発者が AWS と Azure のクラウドリソースをより手軽にシンプルに管理できるようにしています。この実現において Terraform は、チームのデータセンターと AWS リソースの管理方法に適合したワークフローおよび構成の基本要素を提供します。

各種のクラウド内やベアメタルと仮想ホストを収容している内部データセンターにインフラを構築する際、チームメンバーは自分たちでクラウドリソースを所有して稼働する方法と、インフラチームが管理する環境内で運用する方法のいずれかを選択できます。

“データセンター内のホストを含め、弊社のホストの大半は Terraform で管理されており、Azure や AWS といったプラットフォームの種別に関係なく同じように構築されています” と、Brown 氏は述べ、次のように続けました。“Terraform を利用して、開発チーム向けのセルフサービス型ビジネスモデルを構築しました。チームメンバーは Terraform のインフラ構成テンプレートを使用して、必要なものを何でもプロビジョニングできます。または、私たちにプロビジョニングを頼んで任せることも可能です。一元的なゲートキーパーチームによる確認と承認のプロセスがなくなり、開発者間の摩擦が減りました”

Consul の Key-Value ストア機能は、サービスのオンボーディングとオフボーディングの処理に欠かせない重要なツールです。このプラットフォーム非依存のディスカバリソリューションは、数千ものホストに展開されてサービス登録を一元化するほか、GitHub のさまざまなクラウドやプライベートデータセンター内のホストにわたってサービスディスカバリとメンテナンスステータスの開始も自動化します。こうすることでサービスへの接続が大幅に簡素化され、より速く行われます。

これと同時に、手作業で行っていたシークレット管理が Vault で代行されるようになり、アプリケーションが展開されると常にシークレットが自動的に挿入されて、証明書とアクセス制御が管理されるようになりました。Vault の公開鍵基盤 (PKI) は、多数の Kubernetes クラスタをコマンドラインツールで管理していた会社にとって便利な存在です。Vault でシークレットを動的に生成できるため、オペレーションを自動化して 1 日に大量のシークレットリクエストを効率良く処理できるようになり、貴重な時間とリソースの節約につながりました。自社製シークレットソリューションを業界標準に引き上げる作業にチーム総出で取り組むかわりに、より大きな戦略プロジェクトに取り組めるようになりました。

また、GitHub のセキュリティチームは Vault の証明書管理を使用してアクセス制御を管理し、然るべきチームにのみ特定のシークレットセットへのアクセス権が付与されるようにしています。

” HashiCorp のソリューションは直感的で使いやすく、初期設定が完了すれば自動で機能してくれます。ですから私たちは、より価値の高い戦略や業務に専念することができるのです”

SCOTT SANDERS 氏  
インフラ担当 VP

## セキュリティ、効率性、信頼性を優先させる

Sanders 氏によると、シームレスなファーストクラスの開発者エクスペリエンスを提供することは、社内開発のソリューションに不慣れな新米エンジニアをオンボーディングする上で特に重要です。テクノロジーは利用しやすく、スキルセットは業界内で転用できるものであるため、チームは少ないコストと時間で開発者をスピーディーに育てることができます。これは同社のコンプライアンスとセキュリティ体制の強化にも役立ちました。主要な接続ポイントと機密データの保護が自動化されたためです。このことはプラットフォームに大量の個人情報と専有データを収容している企業には非常に重要なことです。

HashiCorp ソリューションの導入は、同社の日々のオペレーションを大幅に改善させました。“これまで新しいサービスを展開する際は、リポジトリを分離するために大量のプルリクエストが必要で、ホスト 1 つ 1 つに 1 回以上のプロビジョニングコマンドが必要になることもよくありました” と、Sanders 氏は述べます。“新しいサービスを展開するために手作業のステップを 30 分かけて実施し、ロードバランシング構成を再構築していましたが、Consul のテンプレートと組み込みのロードバランシング構成を使用することで、今はこの作業に 1 分もかかりません”

Terraform から同様のメリットを得られたと、Brown 氏は付け足します。“HashiCorp の導入前は、新しいアプリケーションまたはサービスのプロトタイプを作成し開発する際、最低限の製品を完成させるだけでも、ホストへの展開からロードバランシング、シークレット挿入まで数日かかっていました” と述べ、”Terraform の導入により、このプロセスに 1 時間もかからなくなりました。数百もの内部の主要サービスを追跡する間に節約される時間は大きなボーナスであり、より効果的なエンジニアリングを可能にします” と、続けました。

GitHub 社にとってのもう 1 つのメリットは Terraform の拡張性です。Terraform を拡張し、クラウド環境ですでに使用しているワークフローとフレームワークを使用して、オンプレミスデータセンター内のインフラを管理できます。チームはこの強力な機能を GitHub エンジニアリングの残りの部分に取り入れています。

“HashiCorp ソリューションの導入により、OSS ツールの広大なエコシステムを見回して GitHub の問題を解決できるようになりました。それらをソリューションにまとめたものが、弊社のオペレーションと GitHub でのインフラ構築方法の基盤になっています” と、Sanders 氏は述べています。

## 結果



複数クラウド環境に対応する、最適な製品を組み合わせた  
オープンソースソリューションによる標準化



開発者のオンボーディングとトレーニングにかかるコストと労力の削減



数百のサービスと数千のノードにわたってサービスディスカバリと  
シークレット管理を自動化



ロードバランシング構成にかかる時間を 30 分から 1 分未満に短縮

## ソリューション

GitHub 社は HashiCorp の Terraform、Consul、Vault を使用して、クラウドおよびオンプレミスの仮想インフラを迅速にプロビジョニングし、サービスディスカバリーを数百ものサービスにわたって自動化し、シークレット管理を各種のプラットフォームおよびインスタンスにわたって合理化しています。

## GitHub 社パートナー



Git Systems、ストレージ、SRE、コンピュート基盤、DC スペース  
Scott Sanders 氏は GitHub 社のエンジニアリング担当バイスプレジデントとして、インフラストラクチャチームとサイトリライアビリティチームを率いています。2013 年に入社し、グローバルデータセンターおよびエッジネットワークの構築と Kubernetes への移行に携わりました。同氏が構築したツールとベストプラクティスにより、GitHub の速度と可用性が支えられています。

### Scott Sanders 氏

インフラ担当 VP



Aaron Brown 氏は GitHub 社のインフラ担当エンジニアです。2016 年の入社以降、GitHub の数百もの相互接続サービスをホストするインフラの構築に携わってきました。GitHub に入社する前は、EdTech および E コマース業界のインフラチームを率いていました。

### Aaron Brown 氏

インフラストラクチャ担当ソフトウェアエンジニア

# テクノロジースタック

- **インフラ:** ベアメタル & VM (データセンター、AWS、Azure 内)
- **コンテナランタイム:** Docker
- **オーケストレータ:** Kubernetes
- **CI/CD:** GitHub Actions
- **データサービス:** MySQL、ElasticSearch、Kafka、Git
- **バージョン管理:** GitHub
- **プロビジョニング:** HashiCorp Terraform
- **セキュリティ管理:** HashiCorp Vault

