

グローバルバンキングでDevOpsを加速

お客様事例

グローバルなバンキングシステムでソフトウェアの品質向上、デプロイのスピード向上規制の厳しいコンプライアンス要件を満たすためにJFrogはどのようにサポートしたのか

アジア太平洋地域で数十億ドル規模の金融機関であるこの国際銀行では、毎日多額の資金を預かっています。複数のソフトウェア開発チームがWeb、モバイル、POSなどのプラットフォーム向けのフロントエンドとバックエンドのアプリケーションを提供しています。この規制の厳しい業界では、コンプライアンス違反が数百億ドルの賠償金問題に発展することや、経営権の損失にまで及ぶ可能性もあります。

“約3,500人のアクティブユーザーと、110TB以上蓄積されたデータはさらに毎月3TBのペースで増え続けています。”

この銀行では5,000人近い開発者、QA、運用エンジニアが利用するさまざまなプログラミング言語、パッケージタイプ、ビルドエコシステムツールに対応しながら、一貫性を保つためのベストプラクティスの共通開発パイプラインを通じて、組織全体のソフトウェア作成のモダナイゼーションを目指しています。これらの方法を実現するためには、銀行のスタッフや州の規制当局によるフォレンジック分析のための複数年にわたるトレーサビリティを可能とし、信頼性が高く、安全で、コンプライアンスに準拠した結果を生み出す必要があります。同社ではユニバーサルリポジトリマネージャーのJFrog Artifactoryでパッケージ、バイナリ、メタデータの効果的な管理、そしてJFrog Xrayを活用することでデプロイされたすべてのアプリケーションに既知の深刻な脆弱性がないことを保証することで、これらの目的を達成しました。

“XrayはArtifactoryと密接に連携しており、銀行内部に流入する多数の脆弱性を含むイメージをブロックできるという利点があります。”

このモダナイゼーションの成功によって、バイナリ管理をオンプレミスからパブリッククラウドという費用対効果の高いコンピューティング環境に移行し、厳格なデータ保持義務に対応するために使用頻度の低いデータをクラウドアーカイブストレージへの移行も実現しました。

業界

金融業・銀行

結果

パッケージとバイナリの効果的な管理によりJFrogはこの国際銀行で以下を可能にしました。

- 外部ネットワークとのトラフィックを削減しパッケージとアーティファクトを共有することでデプロイサイクルの高速化を実現
- デリバリ頻度を高め、アプリケーションのデプロイとアップデートを日次実行
- DockerとHelmのプライベートレジストリを使用してクラウドネイティブ開発を促進
- 深刻な脆弱性を持つアプリケーションが本番環境にデプロイされないようにブロック
- ライセンス、QA、データ保持に関する規制に準拠
- ダウンタイムなしでクラウドに移行でき拡張性と最適なOpEX/CapEXバランスを実現

会社概要


30,000

人以上の従業員


8 million

人以上のお客様


Top 200

フォーブス・グローバル2000の
世界大手上場企業で(2021年)


\$600 B

以上の運用資産
(米ドル換算)

ソリューション

JFrog Artifactory

JFrog Xray

JFrog製品を使いどのように成功したか

モダナイゼーション以前は、ソフトウェア開発チームがそれぞれ独自のツールセットを採用していたため、品質を検証してリリースするためにポリグロット方式が必要でした。

同社は企業全体でDevOpsベストプラクティスを1つにまとめソフトウェア開発プロセスを最新にしました。また品質を向上させ深刻な脆弱性を排除すると同時に継続して増加するデスクトップやデバイスに対してより頻繁なソフトウェアアップデートの配信を目指しました。

このような運用でしばらくはうまく続けていましたが、すぐにデータ量が急激に増加し、老朽化したオンプレミスへの負担が膨大になりました。同社は手頃な価格でデータをアーカイブしアクティブな運用をクラウドに移行する手段を必要としていました。

この状況下でJFrogがどのようにして同行に成功を導いたのかをご紹介します。

信頼できる唯一のソース

JFrog Platformのインストール後、経営陣は銀行内のすべてのシステムユーザーにArtifactoryを活用してパッケージ、アーティファクト、バイナリを保持するように義務付けました。JavaとJavaScriptが主な言語ですが、Artifactoryは最終的にMaven、Docker、npm、Gradleなど20以上の異なるパッケージタイプでも活用されるようになりました。開発チームはArtifactoryの何千ものリポジトリを通して、銀行組織全体で独自のパッケージを共有しています。

「全てのユーザーはArtifactoryを使用してバイナリを保存する必要があります」

外部リポジトリ用のプロキシ

外部ネットワークのトラフィック削減のために、システム管理者はArtifactoryにリモートリポジトリを作成し、Maven Central、npm、Docker Hubなどのすべての外部リポジトリリソースをプロキシしました。これにより接続が遮断された場合でも、頻繁に使用される依存関係はローカルへ継続してアクセスが可能であり、外部リポジトリにプッシュされた不適切な依存関係や悪意のある上書きを防止できます。

「私たちはすべての外部リポジトリのプロキシとしてリモートリポジトリを利用しています。npm、Maven Central、Docker Hubをプロキシしています」

不変なビルドのプロモーション

ビルドされたすべてのソフトウェアは、ビルド、検証、そして最終的にリリースという3段階の評価をパスしなければなりません。ビルドがテストをパスした後、その不変なバイナリとメタデータはあるステージのリポジトリから次のステージのリポジトリにプロモートされます。品質が確認され、最終的なリリースリポジトリにプロモートされたビルドだけが本番環境にデプロイされます。

「当社ではビルドも上書きしないというポリシーがあります。変更を加えるたびに新しいビルドをプッシュする必要があります」

→

「検証段階に進んだビルドはテスト環境でテストされます。100%問題がなければ本番環境に移行します」

「すべてのデプロイはリリースリポジトリを通して行われます。このプロセスは必須ですべてのチームが行う必要があります」

ロールベースのアクセスコントロール

システム管理者はJFrog Platformのきめ細かいアクセスコントロールを使用してサービスアカウントを管理し、権限に応じたりポジトリのアクセス制御を行います。

「システム管理者は主に公開、プロモート、デプロイする権限があり、それが全てです」

「開発者はプッシュしてビルドし、プロモートして検証するだけです。サポートは検証からリリースにプロモートできリリースしたものからしかデプロイできません」

ビルドエコシステムとの統合

集中型ビルドシステムは採用していません。開発チームは独自のCI/CDツールを使い、それぞれ各自のサーバーで運用しています。JFrogプラグイン、JFrog CLI、REST APIを活用して多様なエコシステムのすべてのツールでビルドとメタデータを集中型Artifactoryにあるリポジトリにプッシュできます。

開発チームの約80~85%がJenkinsを利用しており、各アセットには独自のJenkinsやBambooなどがあります。

クラウドネイティブ開発を加速

ArtifactoryがDockerとHelmをネイティブサポートしていることが、コンテナベースのマイクロサービスアーキテクチャを採用する決め手になりました。ArtifactoryのプライベートDockerレジストリを活用して開発チームは組織内でコンテナイメージの配布や共有が可能です。

「今ある110テラバイトのデータ中の50テラバイトはおそらくDockerだけのデータでしょう」

脆弱性のブロック

JFrog Xrayのポリシーとウォッチを活用して、主要なリポジトリをスキャンし既知の脆弱性を含む依存関係のバイナリにフラグを立て、本番環境にデプロイされるのを防止します。

「XrayはArtifactoryと緊密に連携しており、内部に侵入する多数の脆弱なイメージをブロックできるメリットがあります」

高可用性によるスケーリング

高可用性ArtifactoryとXrayのデプロイを通じて、ミッションクリティカルなリポジトリマネージャーからのレスポンスを維持しセルフホスティングで負荷分散された冗長インスタンスのセットを採用しています。

「約3,500人のアクティブユーザを持ち、110TBを超えるデータ蓄積量は毎月3TBのペースで増え続けています」

アーティファクトのコールドストレージ

3年間の運用結果では、同行は110TBを超えるパッケージ、アーティファクト、バイナリ、メタデータをArtifactoryのリポジトリに蓄積し、毎月さらに3TBずつ増え続けていました。銀行の規制要件では、すべてのデータは長時間保管することが求められています。これだけのデータ量を毎日インデックス化すると、特にXrayスキャンでは完了までに長い時間を要し、生産性に影響します。

ArtifactoryのArtifact Cold Storage機能を活用することで、銀行は80TBの古いデータ（全体の70%以上）を低コストのAWS Glacierクラウドストレージにアーカイブしながら継続的なトレーサビリティを維持できました。わずか30TBのアクティブなデータのみでArtifactoryとXrayのパフォーマンスが回復し、Artifactoryを通じて過去12ヶ月も手つかずだった全データの自動アーカイブをスケジューリングしました。

「わずか1年前には月に2TBしか増えてみませんでしたが、今では3TBです。コンテナがそれを後押ししています」

「私たちは厳しく規制された業界で、極めて厳しい管理・制御をしています。また、フォレンジックな用途で少なくとも7年間はデータを保持しています」

「削除や上書きを一切許可していないため、アーティファクトのツリーストレージの容量が増大し、コンポーネントグラフも巨大化しています。Xrayはデータの「ライブ」サブセットが現状のコールドストレージにあるものと比べて極めて小さいため、パフォーマンスが向上します」

クラウドへの移行

当初は銀行のデータセンター内のサーバーでArtifactoryとXrayをホストしていましたが、後にクラウドプロバイダーのセルフホスティングに移行しました。JFrog Platformで実現したハイブリッドコンピューティングモデルを活用することで、日々の本番環境への影響を最小限に抑えながら、アクティブなワークロードを新しいクラウド環境に移行することができ、開発チームはこの変更が行われてることにほとんど気づきませんでした。

「クラウド化に伴い、クラウド環境は他社で構築して、構築や保守のスキルとノウハウはすべて自社で行いました。結果迅速な開発サイクルを可能にしました」

「老朽化したハードウェアと限られたディスク容量から柔軟なクラウド環境に移行後、全てを自動でデプロイとビルドが可能になりました」

結果

JFrog Platformを使用してDevOpsのベストプラクティスを実現することでソフトウェア開発プロセスをモダナイゼーションすることができ以下のことが実現しました。

- 全チームのソフトウェア開発スピードが向上
- 本番環境への日次デプロイ
- コンテナ化によるクラウドネイティブ開発の高速化
- ロールベースのアクセスコントロールによるセキュリティポリシーの適用
- 本番環境への高品質なリリース、深刻な脆弱性のブロック
- ソフトウェアコンポーネントのトレーサビリティによるフォレンジック分析
- 低コストな自動アーカイブによる効率化とデータ保持ポリシーの準拠
- 高可用性とクラウドへの移行を実現

この移行の成功により、ミッションクリティカルなソフトウェアアプリケーションの品質を担保しつつ、拡張をサポートすることができるようになりました。



JFrog Japan 株式会社

〒100-0004 東京都千代田区大手町1-9-2 Global Business Hub Tokyo | TEL: 03-4243-1049 | Webサイト: jfrog.com/ja/ | ブログ: jfrog.com/ja/blog/
お問い合わせ: jfrog.com/ja/contact-us/

- JFrogの名称、ロゴマークおよびすべての JFrog製品の名称は、JFrog Ltd.の登録商標または商標です。
- その他、本書に記載されている会社名および製品・サービス名は、各社の登録商標または商標です。
- JFrogは、通知を行うことなく、いつでも該当製品およびサービスの提供、機能を変更する権利を留保し、本書中の誤植または図表の誤りについて責任を負いません。