Industry:
Oil & Gas, Energy

Leading Oil & Gas Corporation Modernizes Software Supply Chain with the JFrog Platform

Scalable Secure Software Supply Chain

Unified Platform for Efficiency

Proactive Risk Management and Innovation

$500B	60K+	50+
Revenue	Employees	Development Sites

> *"With the previous solution, there was definitely more burden on our side to make sure the platform scaled and stayed healthy. We don't want our teams focused on keeping infrastructure alive; we want them focused on enabling developers, and that's where JFrog made a real difference."*

— Senior Project Consultant

Overview

This leading multinational oil and gas corporation modernized its software supply chain by replacing Sonatype with JFrog to achieve stronger security, better developer experience, and a scalable, SaaS-based foundation for its software artifact management. The DevOps and Security teams ultimately selected the JFrog Platform for its key differentiations like strong tech integrations, curated open-source protection, container-native scalability, and ability to unify security and development workflows in a single platform.

The global energy leader operates at massive scale – supporting thousands of developers as they continue to scale, and internal teams across secure, highly regulated environments. Its internal development foundation team is responsible for enabling secure software delivery across the enterprise, with a strong focus on open source governance, developer experience, and supply chain security.

The organization manages a large GitHub footprint, supports internal developer portals, mobile development teams, cybersecurity products, and is rapidly expanding its use of AI/ML workflows that require secure and scalable artifact management.

Challenge

Before adopting JFrog, the company relied on Sonatype for filtering open source packages, but faced increasing limitations as their solution could not handle the growing complexity and constant evolvement of the AppSec threat landscape.

Key challenges and pain points include:

Challenge	Pain Point
Limited artifact management capabilities	The existing environment lacked a centralized, enterprise-grade way to manage, store, and govern build artifacts across multiple teams, creating fragmentation, operational risk and inability to scale.
Need for controlled open-source consumption	The organization required a proxy-based approach to control and curate open source packages before they entered internal environments, which their prior tooling could not fully support at enterprise scale.
Poor scalability of the incumbent platform	The existing solution, Sonatype, was built on a VM-based architecture that placed the burden of scaling and availability on internal teams, making it difficult to support growth and increased developer demand.
Weak GitHub integration and ecosystem gaps	The prior tools did not deeply integrate into GitHub-native workflows and lacked support for key ecosystems such as Python (PyPI), limiting developer efficiency and security visibility.

High operational overhead	Maintaining and scaling the incumbent platform required significant internal effort, diverting time away from higher-value work like developer enablement and security strategy.
Developer experience friction	Developers were dissatisfied with the usability and performance of the existing tools, which reduced adoption and created friction in daily workflows.
Inability to unify security and development workflows	The existing toolset forced teams to manage security, artifact management, and governance across disconnected solutions, limiting visibility and slowing response times.
Growing need to support AI/ML and modern workloads	As the organization expanded into AI/ML and modern application development, the prior platform lacked the flexibility and architecture to securely manage new artifact types and workflows.

The incumbent platform, Sonatype, had been effective at addressing early proxy and firewall needs, but could not evolve to support modern, cloud-native development at enterprise scale.

Solution

With over 80% of Fortune 100 companies partnering with JFrog to secure their software supply chains, it's not surprising that many enterprise grade energy companies ensure increased release velocity and enhanced application security with JFrog.

Key advantages of the JFrog Platform include:

✓ **Single, unified software supply chain platform**
JFrog delivered a comprehensive platform that brought together artifact management, open source governance, and application security into a single, coherent system, eliminating the need to stitch together multiple point products and reducing architectural complexity across the enterprise.

> The JFrog Platform provides us with a full suite of capabilities… it’s going to allow us to retire a few other things internally.
>
> — Project Engineer

✓ **Stronger GitHub-native security integrations**
The platform provided deeper, more secure integrations with GitHub than incumbent or competitive tools, enabling identity-based authentication, native pull request feedback loops, and in-context security insights that aligned naturally with existing developer workflows without requiring process changes.

> What really surprised us was how strong the GitHub integrations were. That became a differentiating factor for us compared to what we were seeing from Sonatype. The tighter and more secure integrations with GitHub were a big part of why we moved in this direction.

- ✓ **Enterprise-grade artifact management capabilities**
  JFrog offered robust, production-ready artifact lifecycle management, including secure binary storage, version control, promotion workflows, and traceability across build and deployment stages, addressing a significant functional gap that existed in the previous toolchain.

- ✓ **Cloud-native architecture built for scale**
  JFrog Artifactory integrates with Kubernetes to provide comprehensive solutions for managing containerized applications throughout the lifecycle. Built from the ground up on containerized, Kubernetes-native infrastructure, JFrog enabled elastic horizontal scaling, improved resilience, and high availability while shifting the responsibility of backend performance, uptime, and capacity planning away from internal platform teams.

- ✓ **Proactive open source protection through curation**
  JFrog Curation enabled policy-driven, preemptive control of open source packages by enforcing approval and blocking mechanisms before dependencies entered development pipelines, transforming the organization's approach from reactive vulnerability response to proactive supply chain defense.

- ✓ **Superior developer experience and adoption**
  The platform delivered a more intuitive, consistent, and developer-friendly user experience, reducing friction in daily development activities, increasing trust in security tooling, and driving higher levels of adoption across engineering teams.

- ✓ **Context-aware security across the SDLC**
  JFrog's Advanced Security capabilities correlated risk data across source code, build processes, and deployed artifacts, providing richer, more actionable intelligence than isolated scanners and helping teams prioritize remediation based on real-world impact rather than static severity scores.

- ✓ **Strategic ecosystem integrations for modern workloads**
  The company evaluated JFrog's strong ecosystem integrations and partnerships as a signal of long-term platform viability, particularly in emerging areas such as AI and machine learning, where secure management of models, datasets, and package dependencies is becoming increasingly critical.

- ✓ **Tool consolidation and platform simplification**
  By adopting JFrog as a central platform, the organization was able to plan the retirement of multiple overlapping internal tools and third-party products, simplifying governance, reducing licensing and operational costs, and creating a cleaner, more supportable enterprise architecture.

- ✓ **SaaS-first delivery model aligned to enterprise priorities**
  The SaaS deployment model aligned with the organization's strategic objective to minimize internal platform operations work, enabling teams to focus on higher-value activities like developer enablement and security improvement while receiving automatic upgrades and new capabilities without disruption.

Results

JFrog delivered immediate operational relief and long-term scalability, giving Development, Operations and Security teams a unified, secure, single source of truth throughout the SDLC from coding through distribution and into runtime. After the deployment of JFrog Artifactory, JFrog Xray, JFrog Advanced Security, and JFrog Curation, the organization realized the following benefits:

Architectural Transformation and Tool Consolidation

The organization successfully executed a structured enterprise initiative to migrate away from legacy systems toward a JFrog-first architecture. By replacing old firewall and proxy functions with JFrog-native capabilities, the company was able to retire Sonatype entirely. This transition was managed progressively to ensure the continuity of service and repository migration without disrupting development velocity. Furthermore, while the organization initially validated the platform via self-hosted proof-of-concepts, they ultimately standardized on the SaaS-based JFrog Platform on Microsoft Azure. This shift eliminated the overhead of infrastructure maintenance, patching, and scaling, allowing the internal teams to focus on core business value rather than backend management.

> We don’t want to be in the business of constantly tuning infrastructure just to keep up with growth. With the SaaS model, JFrog is able to handle scaling the backend instead of us – and that’s a huge win for us.

Centralized Governance and Open Source Control

A cornerstone of the new strategy was the enterprise rollout of JFrog Curation as a standardized control plane. By making Curation a mandatory enterprise control point for all open source dependency consumption, the organization established centralized, policy-driven governance. This ensures that only approved packages enter the build pipelines, providing necessary auditability and compliance visibility. Additionally, the platform now serves as the foundation for centralized governance across the entire lifecycle, standardizing how artifacts move through environments by enforcing consistent promotion rules, retention policies, and approval workflows.

> We’re using [JFrog] Curation as a critical path. It’s not optional for us – it’s a mandate. That’s the capability that lets us move away from the firewall and proxy model we had before and fully replace what we were doing with Sonatype.

Developer-Centric Security Integration

To improve security without hindering productivity, the organization integrated JFrog’s Frogbot directly into GitHub-native workflows. This allows developers to receive real-time security feedback on pull requests and commits, enabling them to remediate vulnerabilities within their primary development environment. The rollout of JFrog Advanced Security followed a phased approach, starting with a controlled pilot to validate usability before expanding to thousands of developers. This strategy ensures that security policies and internal enablement programs mature alongside the scaling user base.

> Being able to surface security information directly inside the platforms our developers already live in has made a real difference. Instead of chasing people to fix things, we're meeting them exactly where they work, and it's making adoption much easier.

Future-Proofing and Expanded Use Cases

The implementation has expanded beyond traditional application components to address critical gaps in Infrastructure-as-Code (IaC) security. By using JFrog to scan Terraform and other IaC assets, the company has brought consistency to areas that were previously unmanaged or inconsistently governed. Looking forward, the JFrog Platform is now positioned as the foundational layer for AI/ML artifact governance. This ensures that machine learning models, data packages, and experimental artifacts are managed with the same rigor and security standards as any other software component in the enterprise.

Why Energy Companies Trust JFrog for Application Security

In choosing JFrog, this large energy company not only solved critical scalability and security challenges but also streamlined its entire software delivery process under one unified platform. This strategic consolidation empowered teams to move faster, reduce operational complexity, and maintain stronger control over their software supply chain.

With JFrog's robust, scalable SaaS solution, the company is well-positioned to support ongoing innovation and growth while confidently managing risk in today's fast-evolving technology landscape.

See for yourself how the JFrog Platform can help streamline and secure your software solution by taking an **online tour** or **scheduling a demo** at your convenience.

v1.0240129

ABOUT JFROG

JFrog empowers thousands of DevOps organizations globally to build, secure, distribute, and connect any software artifact to any environment using the universal, hybrid, multi-cloud JFrog Platform.

LEGAL STATEMENT

www.jfrog.com
www.twitter.com/jfrog
www.facebook.com/artifrog/
www.linkedin.com/company/jfrog-ltd