

# DER GUIDE ZUR KONSOLIDIERUNG VON DEVOPS- UND SICHERHEITSTOOL

Best Practices, Tipps und Ansatzpunkte für die Konsolidierung Ihrer Tools für mehr Effizienz, Sicherheit und Kostenoptimierung.



# Inhalt

|                                                                                                          |    |
|----------------------------------------------------------------------------------------------------------|----|
| Zu viele Tools, zu wenig Nutzen                                                                          | 2  |
| <b>Teil 1:</b> Bereiche für die Tool-Konsolidierung in Ihrer DevOps-Toolchain                            | 3  |
| <b>Teil 2:</b> Fünf Schritte zur Konsolidierung Ihrer DevOps-Tools                                       | 5  |
| <b>Teil 3:</b> Den ROI bestimmen                                                                         | 10 |
| <b>Teil 4:</b> Herausforderungen bei der Konsolidierung von DevSecOps-Tools (und wie man sie überwindet) | 12 |
| <b>Teil 5:</b> Konsolidieren mit JFrog                                                                   | 15 |

# Zu viele Tools, zu wenig Nutzen

Fühlen Sie sich von zu vielen Tools erschlagen? Damit sind Sie nicht allein. Laut der IDC verwenden die meisten Unternehmen zwischen sechs und 20 oder sogar noch mehr Tools, um ihren Softwareentwicklungszyklus (SDLC) zu tracken und zu überwachen. Angesichts der Vielzahl von DevOps- und Sicherheitstools und dem Umstand, dass Unternehmen ihren Entwicklungsteams völlige Freiheit bei der Auswahl von Werkzeugen einräumen, kommt es häufig zu einem Wildwuchs an Tools. Die Daten zeigen, und das verwundert nicht, dass DevOps- und Security-Teams nun eine Konsolidierung ihrer Tools in Angriff nehmen.

Im Rahmen einer Tool-Konsolidierung werden redundante Tools kombiniert oder ein einzelnes Tool für unterschiedliche Funktionen gewählt. Denn zu viele DevOps- und Security-Tools können Workflows und Prozesse ausbremsen und so die Effizienz verringern. Eine Tool-Konsolidierung reduziert die Zahl der Tools, die ein Unternehmen verwendet, um Workflows zu optimieren, Prozesse zu standardisieren und die Effizienz zu steigern.

Gleichzeitig sollen DevOps-Teams auch Sicherheitslösungen in ihren Tool-Stack integrieren. Eine aktuelle Umfrage von Gartner zeigt, [dass 75 % der Unternehmen die Konsolidierung von Anbietern von Sicherheitslösungen anstreben](#), gegenüber nur 29 % im Jahr 2020. [Sicherheits- und Risikomanagement-Verantwortliche](#) sind zunehmend unzufrieden mit der operativen Ineffizienz und der mangelnden Integration eines heterogenen Sicherheits-Stacks", sagt John Watts, VP Analyst bei Gartner. "Infolgedessen konsolidieren sie die Anzahl der von ihnen genutzten Security-Anbieter."

Durch die Konsolidierung mehrerer Tools zu einer Single Source of Truth lassen sich Skaleneffekte erzielen, die im Laufe der Zeit zu einer Reduzierung der Investitions- und Betriebsausgaben beitragen können. Durch die Konsolidierung der Tools können auch

die gesamten Pflichten und Beziehungen zu den Anbietern reduziert werden, was zu einer besseren Übersicht und zu höherer Transparenz über die Ausgaben des Unternehmens führt.

Die meisten Anbieter können allerdings nicht alle gewünschten Features in der benötigten Form und Qualität liefern. Daher sollten Sie zunächst ermitteln, welche Features und Use Cases für Sie am relevantesten sind, bevor Sie Ihr Toolset konsolidieren. So können Sie sicherstellen, dass Sie angemessen konsolidieren und die Funktionen, die Ihre Tools bieten, voll ausschöpfen.

In diesem E-Book führen wir Sie durch:

1. [die Bereiche der Tool-Konsolidierung in Ihrer DevOps-Toolchain](#)
2. [Einen Fünf-Schritte-Plan \(mit einer Checkliste und Best Practices\) für die Konsolidierung Ihrer DevOps-Tools](#)
3. [Wie Sie den ROI bestimmen](#)
4. [Häufige Herausforderungen bei der Konsolidierung von DevOps-Tools \(und wie man sie überwindet\)](#)
5. [Konsolidierung mit JFrog, einschließlich relevanter Kundenbeispiele zur Tool-Konsolidierung](#)

# Teil 1

Bereiche für die Tool-Konsolidierung in Ihrer DevOps-Toolchain

## Teil 1: Bereiche für die Tool-Konsolidierung in Ihrer DevOps-Toolchain

Wenn Sie über eine Konsolidierung Ihrer Tools nachdenken, sollten Sie auch die unterschiedlichen beteiligten Teams mit einbeziehen. Manche Teams verwenden vielleicht genau die Tools, die Sie konsolidieren wollen. Vielleicht planen diese Teams aber ebenfalls eine Konsolidierung, so dass sich Ihre Maßnahmen überschneiden. Achten Sie also darauf, wenn Sie Entscheidungen zur Tool-Konsolidierung treffen. Ziehen Sie die Konsolidierung der folgenden Werkzeuge in Ihrem Tool-Stack in Betracht:

- **Artefakt-Storages:** Unternehmen speichern Artefakte und Komponenten oft an unterschiedlichen Orten, z. B. in paketspezifischen Repositories und S3-Buckets, Versionskontrollsystemen (VCS) und anderen Varianten. Das macht es schwierig, den Prozess der Nachverfolgung und der Updates von Artefakten zu automatisieren.
- **Container-Registries:** Um Inkonsistenzen und Fehler bei der Arbeit mit mehreren Images in verschiedenen Umgebungen zu vermeiden, sollten Teams ihre Image-Management-Lösungen zentralisieren.
- **Software-Sicherheits-Lösungen:** Unternehmen setzen oft mehrere verschiedene punktuelle Sicherheitslösungen ein, die aber Lücken in der Abwehr aufweisen, redundante oder komplexe Resultate liefern und die Komplexität des Managements erhöhen können. Die Konsolidierung von Sicherheitstools kann einen einfachen Überblick über den Zustand Ihrer Lieferkette geben und den gesamten SDLC abdecken.
- **Cloud/Dynamic Runtime Deployment Technologien:** Unternehmen verlassen sich auf solche Lösungen, um Anwendungen schnell und einfach in der Cloud oder on-Premise bereitzustellen. Durch die Konsolidierung der Orte, an denen diese Technologien / Dateien gehostet und gewartet werden, können Teams den Aufwand für die Verwaltung mehrerer Tools reduzieren.
- **CI/CD-Tools:** Der Einsatz von CI/CD-Tools führt zu schnelleren Builds und Deployments, was zu einer schnelleren Entwicklung, kürzeren Feedback-Zyklen und einer schnelleren Bereitstellung der Anwendungen führen kann. Durch die Konsolidierung von CI/CD-Tools können Teams Prozesse leichter automatisieren und verfolgen, um Probleme schneller zu erkennen und Korrekturmaßnahmen zu ergreifen, so dass das Risiko von Downtimes verringert und die Zuverlässigkeit der Anwendungen verbessert wird.
- **Monitoring-Funktionen:** Verwenden Sie eine zentrale Plattform, die verschiedene Teile Ihrer Systeme und Prozesse überwachen kann, um die Geschwindigkeit Ihrer Entwicklung zu erfassen. Das hilft dabei, den Monitoringprozess zu optimieren, Einblicke in Entwicklungsabweichungen zu erhalten, um so die Geschwindigkeit zu steigern, und den Aufwand für die Verwaltung mehrerer Tools zu vermeiden. Beispiele für solche Plattformen sind Datadog oder Splunk.

# Teil 2

Fünf Schritte zur Konsolidierung Ihrer DevOps-Tools

## Teil 2 Fünf Schritte zur Konsolidierung Ihrer DevOps-Tools

Bei der Menge verfügbarer Tools kann es schwierig sein, den Überblick über all die Veränderungen zu behalten, die sich ergeben. Um weiter effizient arbeiten zu können, ist es daher unabdingbar, ein System zu finden, mit dem sich die Neueinführung, Ausmusterung und Aktualisierung von Tools verwalten lässt.

Es gibt verschiedene Ansätze, um die richtigen Konsolidierungsfelder im Unternehmen zu finden. Wie also sollten Sie wissen, welche Tools Sie konsolidieren wollen? Im Folgenden finden Sie einen Fünf-Schritte-Plan mit Best Practices für die Tool-Konsolidierung und Ideen für die Umsetzung.

### 1. Machen Sie eine Bestandsaufnahme Ihrer DevOps- und Sicherheitstools

**Wie Sie es umsetzen:** Starten Sie mit einer umfassenden Bestandsaufnahme, welche Tools derzeit verwendet werden und wie diese Tools eingesetzt werden. Werden sie für die Verwaltung von Quellcode und Versionskontrolle eingesetzt? Zur Verwaltung von Builds und Abhängigkeiten? Oder sind sie auf die Bereitstellung ausgerichtet? Probieren Sie Folgendes aus, um eine Inventarliste zu erstellen:

- ☐ Erstellen Sie ein Inventar aller DevOps- und Anwendungssicherheits-Tools, die derzeit verwendet werden. Geben Sie die Namen der Tools, die Versionsnummern und den Zweck an, den jedes Tool in Ihrem SDLC erfüllt.
- ☐ Identifizieren Sie weitere mögliche Verwendungszwecke oder zusätzliche Funktionen Ihrer derzeitigen Tools, die Sie bisher nicht aktiv nutzen. Zum Beispiel bieten viele Tools mittlerweile integrierte Sicherheitsfunktionen.
- ☐ Entwickeln Sie einen Prozess oder einen Zeitplan für eine regelmäßige Aktualisierung der Inventarliste. Dabei sollte berücksichtigt werden, wenn neue Tools hinzukommen, alte Tools ausrangiert werden und vorhandene Tools aktualisiert werden.

- ☐ Schaffen Sie ein System für die Vergabe von Zugriffsrechten auf das Bestandsverzeichnis. Dabei sollten Sie festlegen, wer auf das Verzeichnis zugreifen darf und wer nur Lesezugriff hat. Diese Empfehlung gilt auch dann, wenn Sie Ihr Inventar in einer einfachen Tabelle führen.
- ☐ Setzen Sie Automatisierungstools ein, um die Liste auf dem neuesten Stand zu halten. Das könnte heißen, Skripte zu schreiben, die automatisch nach neuen Versionen von Werkzeugen suchen oder regelmäßige Checks der Tools auf der Liste planen.
- ☐ Schaffen Sie eine Möglichkeit, die Nutzung Ihrer DevOps-Tools zu überwachen. Das könnte die Erfassung der Nutzung im Zeitverlauf oder die Einrichtung von Alarmen beim Überschreiten bestimmter Schwellenwerte umfassen.



**FROG-TIPP:** Die Erstellung eines Tool-Inventars ist ein sich wiederholender Prozess. Um Ihren Techstack zu optimieren, überprüfen Sie kontinuierlich die genutzten Tools und bewerten, wie sie DevOps-Prozesse unterstützen oder behindern

## 2. Finden Sie heraus, was Ihnen bei Ihren Tools am wichtigsten ist

**Wie Sie es umsetzen:** Überprüfen Sie Ihre Inventar-Liste, um herauszufinden, welche Tools konsolidiert oder ersetzt werden sollen. So können Sie die Schwerpunktbereiche für die Werkzeugkonsolidierung ermitteln:

- ☐ **Feature-Sets:** Vergleichen Sie den Funktionsumfang der DevOps-Tools in Ihrem Bestand, um festzustellen, ob sich Funktionen überschneiden und in einem Tool konsolidiert werden können. Ermitteln Sie außerdem, welche Funktionen besonders wichtig sind und wo Sie eine solide Betreuung brauchen, statt einfach nur eine schnelle Entscheidung zu treffen.
- ☐ **Wartungsbedarf:** Analysieren Sie den Wartungsbedarf für jedes Tool, um herauszufinden, welche davon konsolidiert oder ersetzt werden sollten. Stellt der Anbieter beispielsweise einen SaaS- oder Managed-Service-Ansatz bereit? Wie hoch ist die Supportleistung? Gibt es spezielle Serviceleistungen? DevOps-Teams sollten sich diese Fragen stellen, um besser zu verstehen, wie sie ihre Ausgaben effektiv zuweisen und ihren Wartungsaufwand optimieren können.
- ☐ **Kosten:** DevOps-Teams sollten natürlich auch die Kosten berücksichtigen, aber nicht ausschließlich. Bevor sie eine Entscheidung treffen, sollten die Teams die funktionalen Anforderungen ihrer Prozesse ermitteln und dann eine Kostenanalyse des Ist-Zustands durchführen, um zu entscheiden, ob eine Konsolidierung wirklich die kostengünstigste Option ist. Diese Analyse sollte alle potenziellen Einsparungen berücksichtigen, die sich aus der Konsolidierung ergeben, z. B. weniger Personal, Infrastruktur und andere damit verbundene Kosten. Die Teams sollten auch Risiken der Konsolidierung berücksichtigen, wie z. B. eine geringere Flexibilität.



**FROG-TIPP:** Nicht alle Tools können konsolidiert werden. Und die Konsolidierung braucht oft Zeit, da die alte Infrastruktur möglicherweise nach und nach ersetzt werden muss.

## 3. Entscheiden Sie, welche Bereiche erhalten bleiben, zusammengelegt werden oder verschwinden sollen

**Wie Sie es umsetzen:** Nachdem Sie nun eine Liste an Tools und Prioritäten erstellt haben, ist es an der Zeit zu klären, welche Bereiche am wichtigsten sind, um die Sie entweder zu konsolidieren oder ein spezifisches Tool dafür zu verwenden. Prüfen Sie die folgenden Bereiche, die Ihre Maßnahmen zur Konsolidierung unterstützen könnten.

- ☐ **Automatisiertes Testing:** Identifizieren Sie Lücken oder Überlappungen bei Ihren automatisierten Tests. Überprüfen Sie den Umfang und die Methoden der durchzuführenden Tests. Identifizieren Sie Bereiche, die nicht automatisiert sind, und überprüfen Sie den Umfang und die Methoden der erforderlichen Tests.
- ☐ **Source Control:** Das Management des Quellcode und die Versionskontrolle sind ein naheliegender Bereich, in dem Unternehmen eine Konsolidierung der Tools in Erwägung ziehen sollten, wenn sie dies nicht bereits getan haben. Insbesondere mit dem Aufkommen von GitOps und der Rolle, die es für einen reibungslosen Betrieb von Anwendungen im Produktivbetrieb spielt.
- ☐ **Monitoring und Logging:** Je mehr Überblick Sie über Ihre Prozesse, Infrastruktur und Anwendungen haben, desto besser. Aber zu viele Monitoring- und Logging-Tools generieren Datensilos, was es schwierig macht, Ihre Daten wenn erforderlich komplett zu analysieren.

- ❑ **Konfigurationsmanagement:** Untersuchen Sie, welche Konfigurationsmanagement-Tools eingesetzt werden und wie sie verwaltet werden. Konfigurationsmanagement-Tools wie Chef, Puppet und Ansible können zur Automatisierung der Bereitstellung, Konfiguration und Verwaltung von Infrastrukturen und Anwendungen eingesetzt werden, erfordern jedoch oft eigene Verwaltungslösungen für jedes einzelne Tool, statt ein einziges System für die Einrichtung und Verwaltung aller dieser Tools bereitzustellen.
- ❑ **Containerisierung:** Bei der Bereitstellung von Container-Anwendungen für den Produktivbetrieb gibt es mehrere Elemente - von der Erstellung über die Absicherung und das Hosting bis hin zur Ausführung -, für die jeweils ein potenzielles Tool zur Verfügung steht. Wenn die Containerisierung ein wichtiger Aspekt Ihrer Software-Strategie ist, sollten Sie überlegen, ob eine Konsolidierung in diesem Bereich sinnvoll ist.
- ❑ **Anwendungssicherheit:** Überprüfen Sie, welche Security-Tools Sie aktuell einsetzen, ob Sie mit einem gewissen Kostendruck zu kämpfen haben und wie viele Tools und individuelle Anpassungen erforderlich sind, um einen umfassenden Schutz Ihrer Software-Lieferkette zu gewährleisten. Beachten Sie das Sprichwort "Shift Left to Ship Right".
- ❑ **Artefakt-Management:** Denken Sie darüber nach, was das "Ende" Ihrer Artefaktverwaltung ist. Müssen Sie zum Beispiel transitive Abhängigkeiten und Builds von Drittanbietern in den Griff bekommen? Die Konsolidierung von Artefakten in einem einzigen Repository-Manager bedeutet, dass Sie sich Gedanken über Ihr aktuelles Value Stream Management, Ihre Kuratierungs- und Archivierungsfunktionalitäten machen müssen und darüber, ob Ihre Metadaten verwertbar sind.



**FROG-TIPP:** Es sei noch einmal darauf hingewiesen, dass manche Tools zusammen besser funktionieren als für sich allein. Es kann also sinnvoll sein, mehrere Werkzeuge zu kombinieren, um das gewünschte Ergebnis zu erzielen.

#### 4. Greifen Sie auf Anbieter zurück, die Tools zusammenführen, um Ihr Software-Updates zu beschleunigen:

**Wie Sie es umsetzen:** Sie können nicht alles konsolidieren, aber ein ganzheitlicher Ansatz kann Lücken schließen und Tools, Prozesse und Workflows miteinander verbinden. Prüfen Sie die Roadmap der Anbieter, um sicherzustellen, dass sie Ihren Anforderungen entspricht. Prüfen Sie, ob Ihr Anbieter Tools mit den folgenden Funktionen anbietet:

- ❑ Umfangreiche OOTB-Integrationen
- ❑ Robuste REST-APIs gepaart mit einer CLI
- ❑ Möglichkeiten zur Automatisierung von Prozessen
- ❑ sowohl als On-Premise-Lösung als auch in verschiedenen Clouds mit konsistenter Benutzererfahrung verfügbar
- ❑ High-Availability und Disaster-Recovery Unterstützung
- ❑ Logging-, Monitoring- und Analysefunktionen
- ❑ Gewährleistung der Sicherheit von Komponenten, Prozessen und Funktionen
- ❑ Unterstützung von SSO und Zugriffskontrolle mit robusten rollenbasierten Zugriffsrechten



**FROG-TIPP:** Vergewissern Sie sich, dass die Tools, die Sie konsolidieren, kompatibel sind, oder suchen Sie eine Plattform, die Sie als Basis für Ihre Toolchain verwenden können.



**FROG-TIPP:** Planen Sie ein Meeting mit der Finanzabteilung, um die Auswirkungen der Konsolidierung auf das Budget zu besprechen, einschließlich möglicher kurzfristiger Mehrausgaben und langfristiger Einsparungen.

## 5. Erstellen Sie einen Plan für die Migration zu einem konsolidierten Set an DevSecOps-Tools:

**Wie Sie es umsetzen:** Nachdem Sie sich nun für einen flexiblen Anbieter entschieden haben und wissen, welche Tools Sie konsolidieren möchten, gilt es einen Plan für die Migration Ihrer Daten aus den einzelnen Tools zu entwickeln. Hier erfahren Sie, wie Sie Ihre Migration am besten planen:

- ☐ Erstellen Sie einen Zeitplan und ein Budget für den Konsolidierungsprozess.
- ☐ Berücksichtigen Sie alle Richtlinien, die sich auf einzelne Werkzeuge oder Prozesse beziehen.
- ☐ Setzen Sie ihren Plan um und überwachen Sie den gesamten Prozess, um die erfolgreiche Konsolidierung Ihrer DevOps-Tools sicherzustellen.
- ☐ Schulen Sie Ihre Mitarbeiter und Benutzer in den neuen konsolidierten DevOps-Tools.
- ☐ Dokumentieren Sie die neue konsolidierte DevOps-Toolchain und warten Sie sie als Teil der IT-Infrastruktur Ihres Unternehmens.
- ☐ Entwickeln und überwachen Sie eine Reihe von Parametern, die die Wirkung Ihrer Tool-Konsolidierung auf das Geschäft genau widerspiegeln.

# Teil 3

Den ROI bestimmen

## Teil 3: Den ROI bestimmen

Im Hinblick auf die Tool-Konsolidierung in DevSecOps ist eine der wichtigsten Überlegungen der ROI. Eine Abwägung bei der Entscheidung für eine Tool-Konsolidierung ist dabei das Verhältnis von Kosten und Zeit.

Kennen Sie schon die Ergebnisse des **Voice of the Customer**-Reports?



**90% der IT-Einkäufer wollen mit weniger Anbietern arbeiten**



**40% sagen es gibt zu viele Preismodelle**



**72% sagen die Preisgestaltung ist zu komplex**

Verwenden Sie diese Formel, um zu berechnen, wie hoch die Kosten für ausufernde Tools sind, bevor Sie sich für die Konsolidierung von Werkzeugen entscheiden:

Lizenzkosten + Infrastrukturkosten + Wartungskosten + Schulungskosten + Integrationskosten

Zusätzlich zu dieser Formel zur Berechnung der Kosten für die Werkzeugflut können Sie mit dieser Checkliste den ROI für die Tool-Konsolidierung ermitteln

- ☐ Gewinnen oder verlieren Sie durch die Konsolidierung von Tools oder den Wechsel zu einer neuen Lösung Funktionen, die wesentliche Anwendungsfälle in Ihrem Unternehmen ermöglichen? Wie hoch wäre der Aufwand, um mit den vorhandenen Tools und dem vorhandenen Setup die gleiche Funktionalität zu erreichen?
- ☐ Verfolgen Sie Metriken wie die eingesparte Überarbeitung und die in neue Funktionalitäten investierte Zeit. Diese Metriken können helfen, ein klares Bild davon zu erhalten, wie viel Zeit und Geld durch die Konsolidierung von Werkzeugen eingespart wird
- ☐ Berücksichtigen Sie den Schulungsaufwand, den die Mitarbeiter für die Nutzung der neuen Tools brauchen. Auch wenn die Schulung der Mitarbeiter auf die neuen Werkzeuge anfangs mit einem gewissen Zeit- und Kostenaufwand verbunden ist, wird sich diese Investition langfristig auszahlen, wenn sie zu einer höheren Effizienz und einer geringeren Anzahl an Tools führt, auf die zukünftige Mitarbeiter geschult werden müssen. Berücksichtigen Sie diese Faktoren, wenn Sie entscheiden, ob sich eine Tool-Konsolidierung für Ihr Unternehmen lohnt.



**FROG-TIPP:** Eine flexible Lösung bringt eher ein positives Ergebnis als eine starre Lösung

# Teil 4

Herausforderungen bei der Konsolidierung von DevSecOps-Tools  
(und wie man sie überwindet)

## Teil 4: Herausforderungen bei der Konsolidierung von DevSecOps-Tools (und wie man sie überwindet)

Bei der Konsolidierung von DevOps- und Security-Tools sollten Sie diese Herausforderungen und Lösungen im Auge behalten, wenn Sie neue Tools implementieren oder die Infrastruktur und Architektur bestehender Tools ändern.



### Herausforderung

Einbindung aller Akteure



### Lösung

Beziehen Sie die wichtigsten Akteure in den Prozess der Toolauswahl und/oder -konsolidierung ein. Zeigen Sie ihnen den Mehrwert der neuen Lösung, für die Sie sich entschieden haben, anhand der Kennzahlen, die Sie mit der ROI-Formel in Teil 3 berechnet haben.



### Herausforderung

Konsolidierung und Auswahl der richtigen Tools



### Lösung

Es gibt eine Unmenge an Optionen, daher ist es wichtig, dass Sie die Tools auswählen, die Ihren Anforderungen am besten entsprechen. Nutzen Sie den Fünf-Schritte-Plan für die Konsolidierung Ihrer Tools in Teil 2 und berücksichtigen Sie den ROI anhand der Formel in Teil 3.



### Herausforderung

Die passenden Werkzeuge miteinander kombinieren



### Lösung

Achten Sie darauf, wie gut sich mögliche Lösungen mit anderen Tools kombinieren und automatisieren lassen, sowie auf die generelle Flexibilität der Lösung. Nutzen Sie den Fünf-Schritte-Plan zur Konsolidierung Ihrer Tools in Teil 2, um zu entscheiden, welche Tools Sie kombinieren oder abhängig von Ihren Anforderungen und/oder Ihrem Budget auslaufen lassen können.



### Herausforderung

Schulung Ihres Teams



### Lösung

Eine Schulung des Teams im Umgang mit neuen Tools ist für eine gelungene Umstellung unerlässlich. Ohne eine entsprechende Einweisung wissen die Teammitglieder möglicherweise nicht, wie sie die Tools effektiv nutzen können, was zu einer höheren Zahl von Fehlern bei den Builds und zu Verzögerungen bei der Entwicklung und Verteilung Ihrer Software führen kann. Indem Sie Ihren Mitarbeitern die erforderlichen Ressourcen und Trainings zur Verfügung stellen, können Sie ihnen den Weg zum gewünschten Erfolg ebnen. Siehe dazu auch Teil 2: Fünf Schritte zur Konsolidierung Ihrer DevOps-Tools



### Herausforderung

Bewältigung der Vorabkosten



### Lösung

Eine gewisse Zeit lang müssen Sie sowohl die bestehenden als auch die neuen oder optimierten Tools unterhalten. Dies kann mit Vorabkosten für neue Lösungen und zusätzlichen Arbeitsstunden für die Einrichtung der neuen Tools und die Migration verbunden sein. Zeigen Sie die langfristigen Kosteneinsparungen und Effizienzsteigerung auf, um diese Ausgaben zu rechtfertigen und eine Freigabe des Budgets anhand der ROI-Formel in Teil 3 zu erhalten.



### Hitachi Vantara

**Problem:** Multisite-Pipelines, isolierte Akquisitionen, inkonsistente Sicherheitsarchitektur

**Ergebnisse:** Einheitliche SSC-Praktiken, schnellere Pipelines, "shift-left" Security-Praktiken

**Lösungen:** JFrog Artifactory, JFrog Xray, Amazon Web Services, Docker Hub

**Mehrwert:** Konsistente Anwendung von Sicherheitsmaßnahmen, Reduzierung der Ausgaben für mehrere Tools



**Problem:** Build-Zeiten, unterschiedliche Plattformen, keine Möglichkeit, den Build-Kontext zu verfolgen oder den CD-Zyklus aufrechtzuerhalten

**Ergebnisse:** Mehrere Repositories, verkürzte Build-Zeiten, schnellere CD-Releases

**Lösungen:** Conan, JFrog Artifactory

**Mehrwert:** Verlagerung des Arbeitszeitaufwands weg von der Verwaltung mehrerer Toolsets und Einhaltung der Budgetanforderungen



**Problem:** Multisite-Pipelines, isolierte Akquisitionen, inkonsistente Sicherheitsarchitektur

**Ergebnisse:** Einheitliche SSC-Praktiken, schnellere Pipelines, "shift-left" Security-Praktiken

**Lösungen:** JFrog Artifactory, JFrog Xray, Amazon Web Services, Docker Hub

**Mehrwert:** Konsistente Anwendung von Sicherheitsmaßnahmen, Reduzierung der Ausgaben für mehrere Tools

Für weitere Erfolgsgeschichten unserer Kunden, besuchen Sie das [JFrog Resource Center](#)

Erfahren Sie im folgenden Abschnitt mehr über Tool-Konsolidierungen von tatsächlichen Kunden und die damit erzielten Erfolge.

### Drei Beispiele aus der Praxis für Tool-Konsolidierungen und deren Mehrwert

Hier erfahren Sie, wie die JFrog Software Supply Chain Platform die Kosten reduzierte und die Skalierbarkeit, die CI/DC-Workflows und die Sicherheitsstruktur für Hitachi Vantara, TomTom und Yunex Traffic verbesserte:

# Teil 5

Konsolidieren mit JFrog

## Teil 5: Konsolidieren mit JFrog

JFrog's **Software Supply Chain** Plattform bietet für viele Unternehmen einen idealen Ansatzpunkt für eine Konsolidierung. JFrog bietet Best-in-Class-Lösungen in drei Bereichen: DevOps, Sicherheit und IoT, die alle in einer einzigen Plattform zusammenlaufen. Unternehmen, die sich für JFrog entscheiden, können von der Konsolidierung in folgenden Bereichen profitieren

- **Paketmanager und paketspezifische Tools:** Konsolidieren Sie Ihr Komponenten-Ecosystem in einer gut verwalteten, automatisierten und ganzheitlichen Lösung, die als Single System of Record für das gesamte Unternehmen dient.
- **Container-Registry und Deployment-Technologien:** Behalten Sie die komplette Transparenz und die volle Kontrolle über Ihre Anwendungen während des Deployments. Die Einbindung Ihrer IaC- und Konfigurationsdateien mit Ihren Containern und Build-Metadaten in JFrog bietet den direktesten und nachvollziehbarsten Pfad für das, was in die Produktion gebracht wird.
- **Sicherheit in der Software-Lieferkette:** Eliminieren Sie punktuelle Sicherheitslösungen zugunsten einer End-to-End-Anwendungssicherheitslösung, die zuverlässige und durchgängige Sicherheit in der gesamten Lieferkette garantiert.
- **Continuous Integration:** Verwalten und orchestrieren Sie alle Ihre Pipelines von einem Ort aus. JFrog liefert eine native, enterprise-grade und auf Sicherheit fokussierte CI-Lösung, die sich auch auf Ihre bestehenden Workflows für "Pipelines für Pipelines" setzen lässt.
- **Infrastruktur, Deployment und Plattform-Automatisierung:** Ersparen Sie sich zusätzliche Tools von Drittanbietern, um Ihre Deployment- und Infrastrukturtechnologien wie Terraform und Ansible zu verwalten. Erstellen und verwalten Sie benutzerdefinierte Workflows von der JFrog-Plattform aus.
- **Edge- und IoT-Verwaltung:** Sie profitieren von einem vereinfachten Ansatz für Geräte-Updates, der auch Ferndiagnose und -überwachung, Flottenmanagement und Sicherheitsmonitoring umfasst.

Mit der JFrog Software Supply Chain Platform profitiert Ihr Unternehmen von folgenden Punkten:

Business Value

Nutzen

TIEFERE DEV + SEC + OPS INTEGRATION

- ✓ Tiefgehende Plattform-, Modul- und Anwendungsintegrationen vom ersten Tag an
- ✓ Gemeinsame Datenbank und einheitliches Datenmodell
- ✓ Geteilte kontextbezogene Informationen mit integriertem Reporting
- ✓ Erweiterte Automatisierungen und intelligente domainübergreifende Policies
- ✓ Integrierte High Availability für unterbrechungsfreien Betrieb
- ✓ Ein erstklassiges weltweites 24/7 Support-Team

Business Value

Nutzen

GRUNDSÄTZLICH FLEXIBEL, OFFEN & ERWEITERBAR

- ✓ Umfangreiche native Unterstützung für Pakete und Dateitypen (30+ Integrationen)
- ✓ Fortschrittliche Multi-Cloud-Funktionalität
- ✓ Zentralisiert oder großflächig verteilt (IoT)
- ✓ Vorab-Integrationen von Partnern "Out-of-the-Box" & ab Tag 1
- ✓ REST APIs für fast alles in Kombination mit der Open Source JFrog CLI
- ✓ Fangen Sie klein an oder skalieren Sie auf die Verwaltung von Millionen von Artefakten und Builds in Ihrem Zeitplan/Budget

Business Value

Nutzen

UMFASSENDE SICHERHEIT, SICHTBARKEIT UND KONTROLLE

- ✓ Erweiterter Einblick in die SBOM
- ✓ Umfangreiche Erfassung und Anreicherung von Metadaten
- ✓ Schnelle Extraktion und Kontextualisierung von SBOM-Informationen
- ✓ Erweiterte Reporting-Funktionen quer durch die Organisation
- ✓ Einblicke und Benchmarks treiben datengetriebene Optimierung voran
- ✓ JFrog's weltweit führende Security-Experten und Ingenieure



Mit JFrog können Unternehmen ihre DevOps- und Security-Tools konsolidieren, um ihre Software-Delivery-Pipelines zu streamlinen und Software-Releases zu automatisieren.

## Fazit: Weniger verwalten, mehr erreichen

Da Unternehmen versuchen, Kosten zu senken und gleichzeitig die Sicherheit der Software-Lieferkette zu verbessern, wird die Konsolidierung von Tools immer mehr zum Gebot der Stunde. Die Konsolidierung von DevOps-Tools reduziert die Komplexität, verbessert die Übersicht und die Sicherheit und ermöglicht den Teams eine bereichsübergreifende Kollaboration, wodurch ein optimierter Workflow entsteht, der die Bereitstellung von Software beschleunigt und die Betriebskosten senkt.

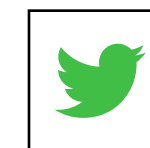
Der Mehrwert und die Vorteile liegen auf der Hand. Vor allem in unserer sich rasant entwickelnden Welt werden nur die Unternehmen überleben und wachsen, die in der Lage sind, ihre Software schnell und verlässlich auf den Markt zu bringen und zu vertreiben.

## Über Jfrog

JFrog hat es sich zur Aufgabe gemacht, eine Welt zu schaffen, in der Software ohne Reibungspunkte vom Entwickler zum Gerät geliefert wird. Angetrieben von der Vision einer "Liquid Software" ist die JFrog Software Supply Chain Plattform ein einheitliches System, das es Unternehmen ermöglicht, Software schnell und sicher zu erstellen, zu verwalten und zu verteilen und dabei sicherzustellen, dass sie verfügbar, nachvollziehbar und vor Manipulationen geschützt ist. Die integrierten Sicherheits-Features helfen auch, Bedrohungen und Schwachstellen zu identifizieren, sich vor diesen zu schützen und sie zu beheben. JFrog's hybride, universelle Multi-Cloud-Plattform ist sowohl als selbstgehostete Lösung als auch als SaaS-Lösung für die wichtigsten Cloud-Service-Anbieter verfügbar. Millionen von Nutzern und über 7.000 Kunden weltweit, darunter die Mehrheit der FORTUNE 100 Unternehmen, verlassen sich auf JFrog-Lösungen, um die Digitalisierung erfolgreich und sicher zu meistern. Wenn Sie einmal einen Sprung nach vorne gemacht haben, gibt es kein Zurück mehr.



[www.jfrog.com](http://www.jfrog.com)



[www.twitter.com/jfrog](https://www.twitter.com/jfrog)



[www.facebook.com/artifrog/](https://www.facebook.com/artifrog/)



[www.linkedin.com/company/jfrog-ltd](https://www.linkedin.com/company/jfrog-ltd)

