WebSphere Hybrid Edition

Tables of Contents

WebSphere Hybrid Edition

Learn more

Overview

Installation

Installation

Help and support

IBM Support Portal

IBM solutions closely align with other industry-leading software providers. While IBM values the use of inclusive language, terms that are outside of IBM's direct influence, for the sake of maintaining user understanding, are sometimes required. As other industry leaders join IBM in embracing the use of inclusive language, IBM will continue to update the documentation to reflect those changes.

Overview

IBM WebSphere Hybrid Edition is designed for on-premises, cloud, and hybrid cloud deployments.

WebSphere Hybrid Edition includes the following WebSphere Application Server editions:

- WebSphere Application Server Network Deployment
- WebSphere Application Server
- WebSphere Application Server Liberty Core

WebSphere Hybrid Edition is the solution of choice for existing on-premises WebSphere Application Server deployments, which are the WebSphere deployments that have powered a good portion of the world's economy for the last 20 years. With WebSphere Hybrid Edition, existing applications can remain in place. WebSphere Hybrid Edition gives you the flexibility to move between WebSphere editions without additional entitlements.

When it is time to transition WebSphere Application Server applications to the cloud, WebSphere Hybrid Edition is also the solution of choice. WebSphere Application Server and WebSphere Application Server Network Deployment include the WebSphere Liberty and Open Liberty options. With its extremely small footprint, industry-leading performance, and full Java™ EE/Jakarta EE and MicroProfile certification, Liberty is an industry standard for deployments in the cloud, whether modernizing existing applications or creating new cloud-based applications.

The solution provides complementary value-add options and features to enhance your WebSphere cloud experience. Highlights include:

- IBM Cloud Transformation Advisor: Helps businesses modernize and migrate their applications from on-premises environments to the cloud or to containers. IBM Cloud Transformation Advisor can identify applications that are good candidates for migration and provide advice and recommendations regarding how to migrate that application.
- IBM Mono2Micro: An AI-driven feature based on IBM Research technology that accelerates and can take the risk out of refactoring existing applications into modern microservices that are ready for cloud deployments.
- WebSphere Application Server Migration Toolkit: A set of Eclipse-based tools for WebSphere migration scenarios including cloud migration, WebSphere version to version migration including WebSphere Liberty, and migration from third-party application servers.
- Also supporting WebSphere Hybrid Edition deployments is the IBM Global Services team, with skills and best practices to help take the risk out of application modernization projects regardless of whether the project spans one application or an entire enterprise.

As WebSphere applications migrate to the cloud, it becomes critical to be able to manage the resultant hybrid environment. IBM is focused on hybrid implementations as the vehicle for clients to maximize their competitiveness and business results related to cloud deployments. WebSphere has been a leader in hybrid deployments, with support for all the major public clouds, Kubernetes deployments, and now optimizations for Red Hat OpenShift that establish Liberty as a solution for Java Enterprise Edition and Jakarta Enterprise Edition (Java EE/Jakarta EE) and MicroProfile deployments to Red Hat OpenShift in private or public clouds. In addition to WebSphere management, WebSphere Hybrid Edition delivers common services enablement to ensure integration and management capability across the IBM software portfolio, most notably with IBM Cloud Pak® solutions.

WebSphere Hybrid Edition is a flexible solution for WebSphere Application Server deployments that can enable organizations to meet current and future requirements. It delivers a broad range of technical portfolio options, the capability to mix and match across options over time without additional purchases, a cloud-friendly Virtual Processor Core (VPC)-based metric, and the latest entitlement options from IBM, including perpetual and subscription options.

- **What's new in WebSphere Hybrid Edition**
  Learn about new features, improvements, any limitations, and other changes in the latest product release.
- **Deprecations**
  Learn about components that were deprecated in this and earlier releases.
- **Removals**
  Learn about components that were removed in this and earlier releases.
- **Licensing for WebSphere Hybrid Edition**
  This document provides information about licensing and entitlements for WebSphere Hybrid Edition.

What's new in WebSphere Hybrid Edition

Learn about new features, improvements, any limitations, and other changes in the latest product release.

For the latest information about new features, improvements, and changes, see the What's New pages in their documentation:

- WebSphere Liberty operator 1.4.2 (released March 2025)
- WebSphere Application Server Liberty 25.0.0.3 (released March 2025)

    - Network Deployment | Base | Core | z/OS
- WebSphere Application Server 9.0.5.23 (released March 2025)
    - Network Deployment | Base | z/OS
- WebSphere Application Server 8.5.5.28 (released February 2025)
    - Network Deployment | Base | z/OS
- IBM Cloud Transformation Advisor 4.1.0 (released March 2025)
- IBM Mono2Micro 24.0.03.1 (released March 2024)
- WebSphere Application Server Migration Toolkit 25.0.0.1 (released February 2025)

Version 5.1

Version 5.1.0 update

Version 5.1.0 Features and enhancements update

- Adds support for Red Hat® OpenShift® Container Platform version 4.18.
- Adds support for Red Hat OpenShift Container Platform version 4.17.
- Adds support for Red Hat OpenShift Container Platform version 4.16.
- Adds support for Red Hat OpenShift Container Platform version 4.15.
- Adds support for Red Hat OpenShift Container Platform version 4.14.
- Adds support for Red Hat OpenShift Container Platform universal base image (UBI) 8.8.
- Adds support for Red Hat OpenShift Container Platform universal base image (UBI) 8.7.

Version 5.1.0 Deprecations and removals

- Removes support for Red Hat OpenShift Container Platform version 4.12.
- Removes support for Red Hat OpenShift Container Platform version 4.10.
- Removes support for Red Hat OpenShift Container Platform version 4.8.
- Removes support for Red Hat OpenShift Container Platform version 4.7.
- Removes support for Red Hat OpenShift Container Platform version 4.6.

Version 5.1.0

Version 5.1.0 Deprecations and removals

- IBM Cloud Foundry Migration Runtime is removed in IBM WebSphere Hybrid Edition 5.1.0. For more information, see the Software withdrawal and support discontinuance announcement.

Version 5.0

Version 5.0.1 Update

Version 5.0.1 Features and enhancements update

- Application modernization tooling updates:
    - Releases new versions of IBM Mono2Micro (21.0.09),IBM Cloud Transformation Advisor (2.5.0), and WebSphere Application Server Migration Toolkit (21.0.0.3).
    - Adds the ability to define custom rules in Binary Scanner and Transformation Advisor to further adapt modernization analysis to your organizational processes.
    - Adds globalization support in 10 languages to IBM Mono2Micro.
- Adds support for Red Hat OpenShift Container Platform version 4.8.
- Replaces installer with instructions for installing or upgrading IBM Cloud Foundry Migration Runtime and [IBM Cloud Transformation Advisor](install-cta.md) directly.

Version 5.0.1 Deprecations and removals update

- IBM Cloud Foundry Migration Runtime is deprecated in IBM WebSphere Hybrid Edition 5.0.1 and might be removed in a future release of IBM WebSphere Hybrid Edition.
- Removes support for Red Hat OpenShift Container Platform version 4.5.

Version 5.0.1

Version 5.0.1 Features and enhancements

- IBM Mono2Micro is updated to version 21.0.06, which includes the following enhancements:
  - Usability improvements in the graph UI to more efficiently handle large amounts of analysis data
  - New ways to filter partitioning recommendations by use cases and runtime call volume
  - An improved method to track and handle unobserved classes
  - Monolith class dependencies in the partitioning views

Version 5.0.0 update

Version 5.0.0 Features and enhancements

- Adds support for Red Hat OpenShift Container Platform versions 4.6 and 4.7
- Adds links to instructions for upgrading IBM Cloud Foundry Migration Runtime and IBM Cloud Transformation Advisor

Deprecations

Learn about components that were deprecated in this and earlier releases.

If a component is listed as deprecated, IBM® might remove it in a subsequent release of the product.

The following table summarizes deprecated components by version and release. Where possible, the table also indicates the recommended migration action.

- Features deprecated in WebSphere Hybrid Edition 5.1.0
- Features deprecated in WebSphere Hybrid Edition 5.0.1

Features deprecated in WebSphere Hybrid Edition 5.1.0

Table 1. Features deprecated in WebSphere Hybrid Edition 5.1.0

Deprecation	Recommended migration action
None	None

Features deprecated in WebSphere Hybrid Edition 5.0.1

Table 2. Features deprecated in WebSphere Hybrid Edition 5.0.1

Deprecation	Recommended migration action
IBM Cloud Foundry Migration Runtime	None

Removals

Learn about components that were removed in this and earlier releases.

The following table summarizes components removed by version and release. Where possible, the table also indicates the recommended migration action.

- Features removed in WebSphere Hybrid Edition 5.1.0

Features removed in WebSphere Hybrid Edition 5.1.0

Table 1. Features removed in WebSphere Hybrid Edition 5.1.0

Deprecation	Recommended migration action
IBM Cloud Foundry Migration Runtime	None

Licensing for WebSphere Hybrid Edition

This document provides information about licensing and entitlements for WebSphere Hybrid Edition.

Important: This Licensing Guide provides supplementary information to assist you in deploying the Programs you have licensed from IBM within your purchased entitlement. Your license agreement (such as the IBM International Program License Agreement (IPLA) or equivalent, and its transaction documents, including the License Information for WebSphere Hybrid Edition 5.1) is the sole and complete agreement between you and IBM regarding use of the Program.

- Listing of licenses by type
- What do you get with your purchase of WebSphere Hybrid Edition, and what is your entitlement?
- License ratio topics
- Reporting on deployment inside and across Red Hat OpenShift clusters
- Differences in license terms

Listing of licenses by type

This product supports the following license metrics as a unit of measure for usage of the licensed software:

- Virtual Processor Core (VPC)

You can learn about license metric definition and guidance at Passport Advantage / Passport Common License Types & Definitions.

These licenses are used when creating instances of the WebSphere Hybrid Edition components, in the **`spec.license.license`** field of each custom resource:

- Full licenses

- Table of license versions

Full licenses

Full licenses provide the option to utilize any of the enumerated and entitled products at the specified ratio consumptions.

To review the license agreements for any of the following full WebSphere Hybrid Edition licenses, click the link for that license:

- WebSphere Hybrid Edition v5.1 Update (L-ZZBG-6V3K4K)

Table of license version

Table 1. Table of license versions

License	Usage	Description
L-ZZBG-6V3K4K	Production or nonproduction	WebSphere Hybrid Edition 5.1 Update

What do you get with your purchase of WebSphere Hybrid Edition, and what is your entitlement?

WebSphere Hybrid Edition offers an enterprise-ready, containerized software solution for modernizing existing applications and developing new cloud-native apps that run on Red Hat OpenShift. The product is supported on both x86, Linux on Power, and Linux for IBM z.

The program contains bundled offerings that include:

- IBM WebSphere Application Server Liberty Core
- IBM WebSphere Application Server Network Deployment
- IBM WebSphere Application Server

The program contains component offerings that include:

- IBM Cloud Transformation Advisor
- IBM Mono2Micro
- IBM Migration Tools

These offerings can be run in containers as a part of the program or as stand-alone deployments of these offerings outside of WebSphere Hybrid Edition. Licensing for WebSphere Hybrid Edition is either perpetual, monthly, or a subscription license. Not all offerings or capabilities of WebSphere Hybrid Edition are supported currently on Linux on Power or Linux on IBM z.

When you deploy bundled offerings (such as WebSphere Application Server, WebSphere Application Server Network Deployment, WebSphere Application Server Liberty Core) under the WebSphere Hybrid Edition license, you must not exceed the maximum entitlement at any time. Deployments can include a mix of different deployed offerings, either standalone, or in WebSphere Hybrid Edition, or a combination of both. Customers can change the deployed offerings at any time as long as they never exceed their maximum entitlement. Deployment of Red Hat OpenShift Container Platform is not required for deployment of bundled offerings within WebSphere Hybrid Edition. Deployment of Red Hat OpenShift Container Platform can be utilized for any offerings that are not bundled with WebSphere Hybrid Edition.

If customers with perpetual license entitlement do not renew Subscription and Support, their support access key expires, and they are no longer able to download product images from the IBM entitled registry (cp.icr,io). They therefore lose access to the product images unless they mirror the product images from the IBM entitled registry to a customer-owned registry (before Subscription and Support lapses) and configure their system to pull from this registry.

Support availability for each WebSphere Hybrid Edition release follows the published support model as described in the announcement letter. For an updated view of which releases are supported and whether fixes are available for each release, review the Support page.

License ratio topics

- License ratio
- What consumes WebSphere Hybrid Edition license entitlements according to the ratio?
- Non-charged entitlements for WebSphere Hybrid Edition

License ratio

Deployed instances of capabilities in WebSphere Hybrid Edition are charged at different rates based on their ratios. An example of ratio use that is provided in the Production and non-production license ratio table is that 1 VPC of WebSphere Hybrid Edition is required for 8 VPCs of WebSphere Application Server Liberty Core deployment for either production or non-production usage.

Entitlements of WebSphere Hybrid Edition that are used in these ratios can be reused in other ratios at any time, as long as the total entitlement is not exceeded. There is no limit to the number of times that entitlements can be used in different combinations.

What consumes WebSphere Hybrid Edition license entitlements according to the ratio?

Table 2. Production and non-production license ratio

Capability	VPC ratio (capability : WebSphere Hybrid Edition)
WebSphere Application Server Network Deployment	1:1
WebSphere Application Server	4:1
WebSphere Application Server Liberty Core	8:1

Note: Production and non-production usage have the same consumption ratios.
To review your WebSphere Hybrid Edition entitlements and to convert them into product entitlements using the Production and non-production license ratio table, use the conversion calculator at IBM Software Central. An IBMid is required; create an account if you do not already have one. After you log in, click Conversion calculators in the main navigation panel.

Noncharged entitlements for WebSphere Hybrid Edition

Deployment ratios apply to the bundled programs within WebSphere Hybrid Edition only. Additional WebSphere Hybrid Edition capabilities or components, such as IBM Cloud Transformation Advisor, IBM Mono2Micro, and IBM Migration Tools, can be deployed without requiring WebSphere Hybrid Edition VPCs to be counted against their deployment.

Reporting on deployment inside and across Red Hat OpenShift clusters

WebSphere Hybrid Edition reporting using the VPC metric

Deployments under WebSphere Hybrid Edition entitlement can continue to be deployed on the same hardware, and in the same VMs as previously measured and reported with PVUs. Recent updates to IBM License Metric Tool (ILMT) now allow it to track deployments of software programs entitled under VPCs. Use ILMT to keep track of software deployed in VMs and entitled under VPCs.

For containers under WebSphere Hybrid Edition VPC entitlements, the license service available to WebSphere Hybrid Edition deployments can be configured to report the deployments of each bundled

offering in each container. It can also report the container size and how that relates—using the VPC-to-license ratio—to WebSphere Hybrid Edition entitlements.

Figure 1. License server report of deployments of each bundled offering in each container

License server report of deployments of each bundled offering in each container

You can also report deployment aggregates for each Red Hat OpenShift cluster to provide a high-water mark of deployment to track against customer WebSphere Hybrid Edition entitlements.

Figure 2. License server report aggregated for each Red Hat OpenShift cluster

License server report aggregated for each Red Hat OpenShift cluster

For more information about the IBM Cloud Platform License Service as used by deployments of WebSphere Hybrid Edition, see IBM License Service.

Differences in license terms

The license terms for WebSphere Hybrid Edition supersede the license terms of the bundled offerings. However, this policy applies only when there is a conflict of terms. Terms that apply to the bundled programs still apply, if not superseded.

- **Tracking and reporting entitlements with IBM License Metric Tool and IBM Licensing Service**
  IBM provides two tools for tracking and reporting licensing information for your IBM entitlements: the IBM License Metric Tool, and the IBM Licensing Service for Kubernetes-based installations.

Tracking and reporting entitlements with IBM License Metric Tool and IBM Licensing Service

IBM provides two tools for tracking and reporting licensing information for your IBM entitlements: the IBM License Metric Tool, and the IBM Licensing Service for Kubernetes-based installations.

IBM License Metric Tool

With License Metric Tool, you can maintain an up-to-date inventory of software assets that are installed in your infrastructure, gather information about your hardware, and ensure license compliance of your enterprise. You can monitor consumption of PVU, VPC, and RVU MAPC metrics by IBM products under full and subcapacity licensing terms.

- **Registering IBM software products with IBM License Metric Tool**
  - Complying with licensing requirements
- **Ensuring IBM software products are correctly tagged and annotated**
  - Adding an IBM product to the software catalog
  - Assigning components to products and Cloud Paks
  - Managing software inventory and metric utilization
- **Generating reports**
  - How to control and report IBM licenses

IBM Licensing Service

- **Setting up IBM Licensing Service**
  - Viewing and tracking license usage
  - Configuring and verifying completeness of License Service
- **Generating reports**
  - Viewing and tracking license usage

Regulatory compliance

You can gain an understanding of the compliance stance for various regulations or standards that can apply to the products and components in WebSphere Hybrid Edition. Clients are responsible for ensuring their own readiness for the laws and regulations that apply to them.

Clients are responsible for identifying and interpreting any relevant laws and regulations that might affect their users. Clients are also responsible for any actions that their users might need to take to comply with these laws and regulations.

To learn about the compliance stance for a product or component, click the appropriate link. If the product does not have a regulatory compliance page, links to relevant pages within the product documentation are provided instead.

WebSphere Application Server Network Deployment

- Regulatory compliance for WebSphere Application Server Network Deployment
- WebSphere Application Server Network Deployment considerations for GDPR readiness
- WebSphere Application Server Network Deployment security standards configurations

WebSphere Application Server Liberty Core

- Regulatory compliance for WebSphere Application Server Liberty Core
- WebSphere Application Server Liberty Core considerations for GDPR readiness

WebSphere Application Server

- Regulatory compliance for WebSphere Application Server
- WebSphere Application Server considerations for GDPR readiness
- WebSphere Application Server security standards configurations

IBM Cloud Transformation Advisor

- IBM Cloud Transformation Advisor considerations for GDPR readiness

IBM Mono2Micro

- Not applicable. IBM Mono2Micro does not collect, store, process, or otherwise use personal data.

WebSphere Application Server Migration Toolkit

- Not applicable. WebSphere Application Server Migration Toolkit does not collect, store, process, or otherwise use personal data.

IBM WebSphere Hybrid Edition considerations for GDPR readiness

This document is intended to help you in your preparations for GDPR readiness. It provides information about features that you can configure, and aspects of the product's use that you should consider to help your organization with GDPR requirements. This information is not an exhaustive list, due to the many ways that clients can choose and configure features, and the large variety of ways that the product can be used by itself and with third-party applications and systems.

Clients are responsible for ensuring their own compliance with various laws and regulations, including the European Union General Data Protection Regulation. Clients are solely responsible for obtaining advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulations that may affect the clients' business and any actions the clients may need to take to comply with such laws and regulations.

The products, services, and other capabilities described herein are not suitable for all client situations and might have restricted availability. IBM® does not provide legal, accounting, or auditing advice or represent or warrant that its services or products will ensure that clients are in compliance with any law or regulation.

Quick links

GDPR Overview

The General Data Protection Regulation has been adopted by the European Union ("EU") and applies from May 25, 2018.

Why is GDPR important?

GDPR establishes a stronger data protection regulatory framework for processing of personal data of individuals. GDPR brings:

- New and enhanced rights for individuals
- Widened definition of personal data
- New obligations for companies and organizations handling personal data
- Potential for significant financial penalties for non-compliance
- Compulsory data breach notification

Read more about GDPR

- EU GDPR Information Portal: https://gdpr.eu/
- IBM GDPR website: https://www.ibm.com/data-responsibility/gdpr/

Links to GDPR information in bundled products and components

- WebSphere Application Server Network Deployment considerations for GDPR readiness
- WebSphere Application Server considerations for GDPR readiness
- WebSphere Application Server Liberty Core considerations for GDPR readiness
- IBM Cloud Transformation Advisor considerations for GDPR readiness
- IBM Mono2Micro does not collect, store, process, or otherwise use personal data
- WebSphere Application Server Migration Toolkit does not collect, store, process, or otherwise use personal data

Getting started moving applications to the cloud

Your company can realize significant benefits by moving valuable Java™ applications from older architectures to cloud environments. Use this guide to help you understand and begin planning to migrate and modernize your applications.

- Understanding application modernization
- Developing a modernization strategy
- Assessing your applications
- Creating an application modernization plan
- Implementing your application modernization plan
- Links to all resources and downloads

Understanding application modernization

Application modernization encompasses a number of different strategies for improving and accelerating your application runtime and development practices. Although there are different starting points and endpoints for each application's journey, the general process involves updating and moving monolithic Java applications from traditional WebSphere Application Server hosting environments into more streamlined, Liberty-based containers in the cloud and Kubernetes environments.

Application modernization is a journey, following these basic steps:

1. Determine a long-term strategy for your overall modernization effort.
2. Assess an inventory of your company's application estate.
3. Develop a plan for each of the applications you want to modernize. Understanding the complexity, cost and criticality of each application helps drive your plan to fit your overall strategy.
4. Implement your application modernization plans.

The modernization plan you ultimately decide on for each application could include any or all of the following steps:

- Runtime modernization: the migration of your application source code to a modern, cloud- and container-optimized runtime. Typically this step means migrating from traditional hosting environment, such as WebSphere Application Server, to WebSphere Liberty or Open Liberty.
- Operational modernization: moving your application to a Kubernetes-based container orchestration platform, such as Red Hat OpenShift Container Platform. You can move to an on-premises environment or a public, private, or hybrid cloud environment, with portability to move between these options.
- Architectural modernization: refactoring your application into individually deployable and scalable cloud-native microservices.

IBM WebSphere Hybrid Edition provides tools that assist at each step in this journey:

- IBM Cloud Transformation Advisor
- WebSphere Application Server Migration Toolkit
- IBM Mono2Micro
- WebSphere Application Server Liberty
- WebSphere Application Server traditional

Even if you only want to containerize your existing traditional WebSphere Application Server applications for cloud or Kubernetes environments (a process referred to as "lift and shift"), you will reap benefits of operational modernization as you can manage all your applications in a single management plane..

To learn more about application modernization, see the following links:

- What is application modernization?
- Modernizing applications to use WebSphere Liberty
- Modernize your valuable Java applications

Developing a modernization strategy

As you develop your modernization strategy, your applications fall into various categories. Applications in the legacy category are the applications that are cost-prohibitive to be modernized or do not provide significant business value. Applications in the strategic category are mission critical applications that run your business and that you plan to invest future development resources.

For your strategic applications, Liberty is IBM's composable, cloud-ready server, which provides support for the latest Java SE, Java or Jakarta EE, and security enhancements. Liberty is poised to run your business applications targeted for your modernization journey.

For legacy applications, use traditional WebSphere Application Server v8.5.5 and v9.0.5 to provide stability through the lifespan of those applications. One strategy is to isolate each application in its own traditional WebSphere Application Server base container and deploy it to your cloud environment. This change affords you operational modernization, so you can manage all your applications in a single management plane.

Assessing your applications

To better understand your applications as you prepare for modernization, here is the primary set of tools with details of the key functionality each provides:

- IBM Cloud Transformation Advisor
  - Scans WebSphere Application Server traditional profiles to inventory your deployed applications.
  - Analyzes applications to help move them from a WebSphere Application Server traditional profile to containerized deployments.
  - Provides complexity ratings and development cost estimates.
  - Provides options for deploying to different cloud runtimes (Open Liberty, WebSphere Application Server Liberty, and WebSphere Application Server traditional).
  - Generates a customized set of deployment artifacts for the selected target runtime.
  - Also evaluates applications running on Oracle WebLogic, Apache Tomcat, or JBoss application servers.
- IBM WebSphere Application Server administrative console - Liberty readiness analysis
  - Runs in the WebSphere Admin Console versions 8.5.5.16+ and 9.0.0.11+.
  - Analyzes selected enterprise applications and their configuration for moving to Liberty.
  - Produces exportable reports that can be shared with teammates
- IBM WebSphere Migration Toolkit for Application Binaries (binary scanner)

    - Command-line tool to analyze applications running in a traditional WebSphere Application Server, Liberty, and other competitive servers.
    - Provides inventory, analysis, evaluation, and configuration reports to facilitate moving an application to Liberty or to a later version of traditional WebSphere Application Server.

Consider the following:

- IBM Cloud Transformation Advisor produces a full view of your application estate, including more comprehensive application and configuration analyses, provides important planning information, and generates customized assets and recommendations for cloud environments.
- The IBM WebSphere Application Server administrative console provides a quick first look toward making your applications Liberty compatible without installing extra tools. This analysis can be run from the enterprise applications view.
- The binary scanner is a good option when you have a few applications to scan or when you do not have access to the application source code. The scanner can quickly provide insights into your applications and their configuration. It generates the same reports as the reports displayed in the admin console and Transformation Advisor.

As you assess each application, determine whether it can be migrated to Liberty, whether to containerize it and deploy it to a cloud, or should it be left out of your modernization efforts until it is decommissioned. Consider the following questions during your assessment:

- Will you retire or replace the application within 3 to 5 years?
- Can you modify the applications source code?
- Is the application critical to your business?
- Does it use technologies that require it to be upgraded before it can be moved to a new platform?

Creating an application modernization plan

Once you know your modernization strategy, which includes both operational and application changes, and your applications are assessed, it is time to plan what to do with each of the applications.

Start by considering the applications that you identified as being strategic. For these applications, various modernization options are available. For your first attempts, plan to work with small, simple applications in order to build your expertise and confidence, then tackle bigger and more complex applications.

The various application modernization options include:

- Runtime modernization.
    - You can choose to move applications to Liberty for use of a right-sized, container-ready application server.
    - As you move your applications to Liberty, it might be necessary to update your application to run on a later Java or Jakarta EE level. Use the WebSphere Application Migration Toolkit to identify and make the necessary changes. Where possible, the tool includes quick fixes to facilitate the changes.
- Operational modernization.
    - You can choose to place the application in a traditional WebSphere Application Server container that can then be managed along side Liberty and other application stacks.
    - You might choose to leave the application running in its current traditional WebSphere Application Server environment, but want to containerize it.
- Architectural modernization.
    - Either during or after runtime and operation modernization, you might decide to refactor your application into microservices. IBM Mono2Micro offers suggestions for refactoring your application and generates microservice code that you can use as a starting point.

- You could decide to build a new replacement application.

Implementing your application modernization plan

As you carry out your plan, consider the following items.

- Runtime modernization.
  - Using IBM Cloud Transformation Advisor to provide application modernization guidance.
    - How do I get an evaluation version of IBM Cloud Transformation Advisor?
    - How do I install IBM Cloud Transformation Advisor on Red Hat OpenShift Container Platform?
    - How do I install IBM Cloud Transformation Advisor in an air gap environment?
    - Can I install IBM Cloud Transformation Advisor in a non-OCP environment?
  - Migrating your applications to new versions of WebSphere Application Server or Liberty
    - What are the system requirements for WebSphere Application Server and WebSphere Application Server Liberty?
    - How do I get an evaluation version of WebSphere Application Server or WebSphere Application Server Liberty?

    During migrations, use a cross-development-team communications mechanism so that different teams can learn from each other as new coding patterns and new techniques emerge.

- Operational modernization.
  - Determine which Kubernetes-based container orchestration platform you want to use. Red Hat OpenShift Container Platform is available for on-premises deployments on Intel, Power, and z/Linux and zCX operating systems; for private clouds; and for public clouds (Amazon EKS on AWS, Microsoft Azure, and IBM Cloud). For more information about the version support for Red Hat OpenShift Container Platform, see the Red Hat OpenShift Container Platform Life Cycle Policy.
  - Download and use Liberty container images. For more information, see Running WebSphere Liberty in a container.
  - Build your containerized application locally on your desktop. You will need the following items:
    1. Your application source code.
    2. Eclipse Integrated Development Environment (IDE) (https://www.eclipse.org/downloads/).
    3. WebSphere Liberty and Java SE 8 (https://www.ibm.com/support/pages/websphere-liberty-developers).
    4. The WebSphere Liberty Plugin for Eclipse IDE installed with WAMT - WebSphere Application Server Migration Toolkit (https://www.ibm.com/support/pages/websphere-liberty-developers).
    5. Java SDK and Runtime 17 (https://developer.ibm.com/languages/java/semeru-runtimes/downloads/)
    6. Podman (https://podman.io).
    7. Access to Red Hat OpenShift (installed OpenShift CLI) and to the public or private registry where to push the containerized application image.

    WebSphere Application Server Migration Toolkit (WAMT) Eclipse plug-in is invaluable to developers making code changes. The changes identified by WAMT match the issues identified in the assessments generated by IBM Cloud Transformation Advisor and the binary scanner. If your developers do not use Eclipse, they can use the assessment reports to make code changes.

  - The WebSphere Liberty operator is available to help you deploy and manage your containerized application in a Kubernetes-based environment. For more information, see Running a WebSphere Liberty operator.
- Architectural modernization.

Use IBM Mono2Micro to help you make decisions about parceling your application into microservices that can be managed separately.

Focus on updating your common code shared across multiple applications, either as shared libraries or packaged within the EAR files. This focus can benefit multiple applications. By moving multiple applications to the latest version of common JAR files, you can eliminate redundancy within your applications.

Links to all resources and downloads

Links to all products, software, tools, and instructional resources mentioned in the previous sections are collected here.

Links to products and components of WebSphere Hybrid Edition:

- IBM Cloud Transformation Advisor
  - Evaluation version: https://www.ibm.com/support/pages/ibm-cloud-transformation-advisor-downloads
- WebSphere Application Server Migration Toolkit
  - Migration Toolkit for Application Binaries download (binary scanner)
  - WebSphere Application Server Migration Toolkit download (source code scanner)
- IBM Mono2Micro
- WebSphere Application Server Liberty
- WebSphere Application Server traditional

Links to other software and tools:

- Eclipse Integrated Development Environment (IDE)
  - WebSphere Liberty Plugin for Eclipse IDE installed with WAMT - WebSphere Application Server Migration Toolkit (see https://www.ibm.com/support/pages/websphere-liberty-developers).
- Java SDK and Runtime 17 (https://developer.ibm.com/languages/java/semeru-runtimes/downloads/)
- Podman
- Red Hat OpenShift (see https://developers.redhat.com/products/openshift/download)
- Red Hat OpenShift CLI (see https://docs.openshift.com/container-platform/4.17/cli_reference/openshift_cli/getting-started-cli.html)
- Red Hat OpenShift Container Platform
  - Amazon EKS on AWS
  - Microsoft Azure
  - IBM Cloud

  See also Red Hat OpenShift Container Platform Life Cycle Policy

- WebSphere Application Server Migration Toolkit (WAMT) application binaries
- WebSphere Application Server Migration Toolkit (WAMT) Eclipse plug-in (see https://www.ibm.com/support/pages/websphere-application-server-migration-toolkit)
- WebSphere Liberty and Java SE 8 (https://www.ibm.com/support/pages/websphere-liberty-developers)
- WebSphere Liberty container images (see Running WebSphere Liberty in a container)
- WebSphere Liberty operator (see Running a WebSphere Liberty operator)
- WebSphere Application Server traditional (see https://www.ibm.com/docs/en/was-nd/9.0.5?topic=90-running-cloud)

Links to instructional resources:

- What is lift-and-shift?

- What is application modernization?
- Modernizing applications to use WebSphere Liberty
- Modernize your valuable Java applications

- **Migrating to Liberty**
  Liberty is the WebSphere container-ready runtime available to run everything from your Java EE monoliths to your Microprofile microservices to your Spring Boot deployable JAR files. Moving to Liberty modernizes both your runtime and operational environment.
- **Migrating WebSphere Application Server traditional versions**
  Move your applications to a lightweight application server for a cloud environment or if you want to take advantage of newer Java SE, Java EE, or Jakarta EE capabilities. However, if there is a reason to perform WebSphere Application Server traditional release to release migrations, refer to the following information.
- **Downloads**
  Access links to downloads for WebSphere Application Server Migration Toolkit and IBM Cloud Transformation Advisor.
- **Additional resources**
  Use these links to documents and resources to help you plan and perform your WebSphere Application Server migrations and modernizations.
- **Frequently asked questions**
  Get answers to frequently asked questions about moving applications to the cloud.

Migrating to Liberty

Liberty is the WebSphere container-ready runtime available to run everything from your Java™ EE monoliths to your Microprofile microservices to your Spring Boot deployable JAR files. Moving to Liberty modernizes both your runtime and operational environment.

Why move to Liberty from WebSphere Application Server traditional?

Liberty is a container-ready Java EE application server that is perfect for containerized environments.

When you adopt a container strategy, Liberty images are available for both Open Liberty and WebSphere Liberty on Docker Hub and Red Hat certified images.

Liberty has one of the fastest startup times of any Java EE application servers, and it has high server throughput to handle your most important workloads.

Open Liberty is the open source Liberty application server that is developed fully in the open at http://openliberty.io. It supports the Java EE 7 and 8, Jakarta EE 9.1 and 10 full profile programming models as well as Microprofile. It is a fast, composable, production-ready, open source Java EE application server.

WebSphere Liberty is built from Open Liberty and contains more features for Java EE 6 web profile to ease some migration scenarios due to different underlying implementations for Java EE technologies like JPA, JAX-RS, and CDI.

Zero migration

After your initial move to Liberty, the Liberty zero-migration architecture means that you can update the product runtime files and continue to use your existing unmodified applications and configuration with no unwanted or unexpected behavior changes.

If you choose to update your application to a later Java EE level to get the latest capability, the migration tools help you with that as well.

Application modernization assessment and detailed analysis

There are two parts to application modernization: runtime and operations. Runtime modernization refers to moving applications from traditional WebSphere Application Server to Liberty. Operational modernization refers to updating your deployment operations to use container orchestration and with new DevOps and GitOps best practices. Using IBM WebSphere Hybrid Edition and Red Hat OpenShift, all your runtimes can be managed with consistency.

The tools described here evaluate your applications to move to Liberty for runtime modernization. There are some differences between traditional WebSphere Application Server and Liberty that takes some migration effort if your application uses older, deprecated technologies or proprietary APIs. For example, WebSphere Liberty supports Java EE 6 web profile (not Java EE 6 full profile). Liberty has Java EE 7 and 8 full profile support, but it does not include optional Java EE technologies such as JAX-RPC and Entity EJB beans. Some of the WebSphere proprietary APIs that were superseded by Java EE APIs were also removed from Liberty. When you have a Java EE application running on Liberty, the zero migration architecture gives you confidence that it is easy to keep it running there.

As described in the Getting Started guide, developing a strategy and plan are an important part of any modernization project. The modernization and migration tools show you the differences and make it easy to understand which applications are affected so that you can plan.

To start your modernization process, there are a number of tools that can help.

- To start your modernization process, there are a number of tools that can help.
    - IBM Cloud Transformation Advisor
    - WebSphere admin console
    - WebSphere Application Server Migration Toolkit (binary scanner)
    - WebSphere Application Server Migration Toolkit (source scanner)
    - Mono2Micro
    - WebSphere Liberty JAX-RPC Conversion Tool for Maven and Gradle
    - Eclipse Transformer

Here is a comparison of the capability available in our major tools. In the table, the letter X indicates support by the indicated product or component for the feature listed in the first column.

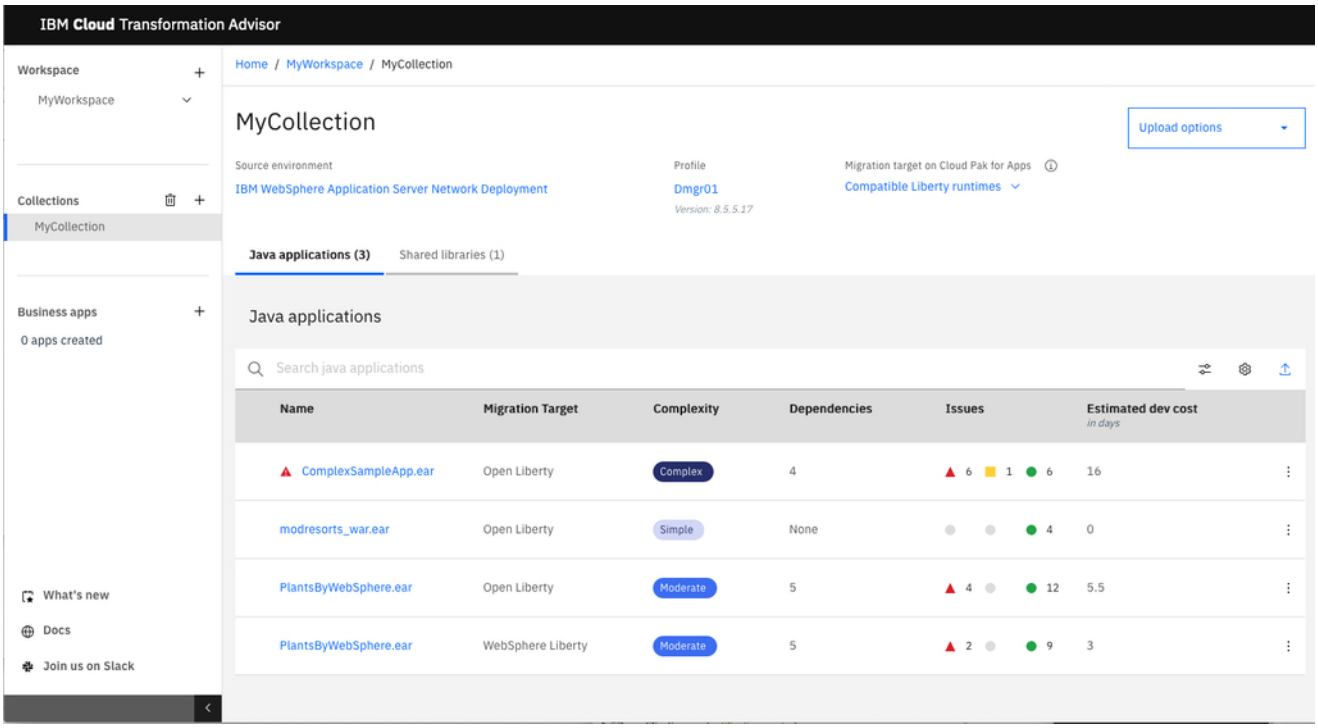
Features	Transformation Advisor	Binary Scanner	Source Scanner	WebSphere Admin Console	Mono2Micro
Enterprise scope	X				
Application detailed migration analysis	X	X	X	X	
Application inventory report	X	X		X	

Features	Transformation Advisor	Binary Scanner	Source Scanner	WebSphere Admin Console	Mono2Micro
Application technology report	X	X	X	X	
Liberty configuration	X	X		X	
WebSphere Application Server traditional configuration	X	X			
Graphical user interface	X		X	X	X
Command line interface		X		X	X
Complexity rating and development costs	X				
Shared library analysis	X				
Container migration artifacts	X				
Interaction with GitHub repositories	X				
Cross product analysis - WAS, MQ, IIB	X				
Business application grouping	X				
HTML and JSON report formats	X	X			
WebLogic, JBoss, Tomcat application analysis	X	X	X		X
WebLogic, JBoss, Tomcat config analysis	X		X		
Containerized implementation	X				X
Source code analysis and generation			X		X
Microservice partitioning					X

IBM Cloud Transformation advisor

To get the large picture of the migration effort for your application estate, start with IBM Cloud Transformation Advisor. The Transformation Advisor data collector scans your entire cell and uploads the results to the Transformation Advisor UI, where you can compare application migration complexities, estimate development costs, and produce migration artifacts for containerized environments. Transformation Advisor reports on runtime modernization for moving from WebSphere Application Server traditional to Liberty, and assists with operational modernization by producing migration artifacts to deploy your workloads in Red Hat OpenShift.

When you cannot move an application to Liberty, Transformation Advisor also gives guidance on moving applications to WebSphere Application Server traditional containers to at least modernize your operations. Remember that Liberty is the preferred container-ready runtime, however.

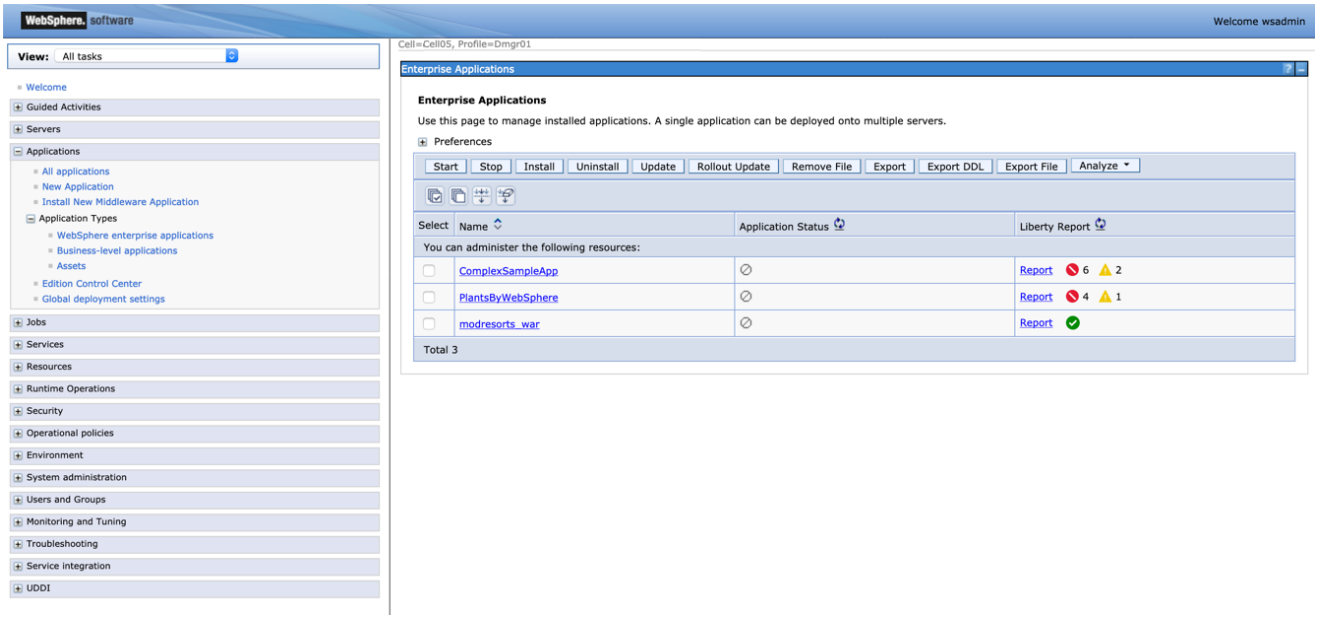


There are several ways to create your Transformation Advisor data collection.

- You can download a data collector from Transformation Advisor.
- You can use the binary scanner to create a Transformation Advisor data collection by using the `--ta` parameter. You can get started by downloading the binary scanner before you install Transformation Advisor.
- Starting with WebSphere Application Server 9.0.5.14, you can use wsadmin to run the data collection that uses the tools that are delivered with WebSphere.

WebSphere admin console

You can analyze applications for their Liberty readiness directly from the WebSphere administrative console when you are running WebSphere Application Server 8.5.5.16 or 9.0.0.11 or higher. Using the Analyze > Run scanner option, you can start the binary scanner to produce a consolidated migration report that can be exported. To learn more, click through this online demo.



WebSphere Application Server Migration Toolkit (binary scanner)

Transformation Advisor and the WebSphere admin console use the same binary scanner tool that you can also download and run easily from the command line, so all of these tools produce migration reports that include:

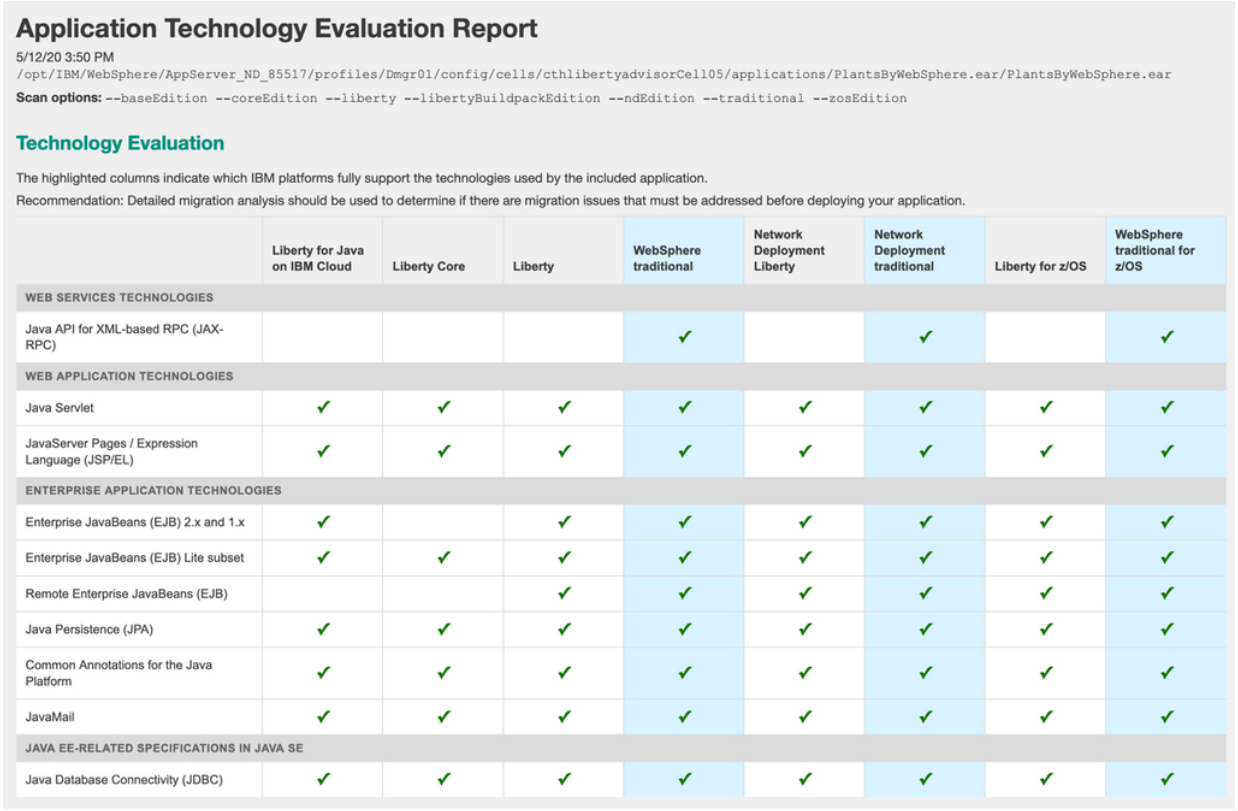
- Application Technology Evaluation Report
- Application Inventory Report
- Detailed Migration Analysis Report
- Liberty or traditional WebSphere Application Server configuration

You can run the binary scanner to produce all four of the reports by default:

```
java -jar binaryAppScanner.jar MyApplication.ear
```

This command creates an Application Migration Report in HTML format. Use the `--help` option to learn the other options including the JSON format option. The various reports are described in the following list.

Application Technology Evaluation Report
: The Technology Evaluation Report shows the editions of WebSphere Application Server that are best suited to run the application. The report provides a list of Java EE programming models that are used by the application, and it indicates whether the application can be supported by Liberty or WebSphere Application Server traditional.



Application Technology Evaluation Report

5/12/20 3:50 PM

/opt/IBM/WebSphere/AppServer_ND_85517/profiles/Dmgr01/config/cells/cthlibertyadvisorCell05/applications/PlantsByWebSphere.ear/PlantsByWebSphere.ear

Scan options: --baseEdition --coreEdition --liberty --libertyBuildpackEdition --ndEdition --traditional --zosEdition

Technology Evaluation

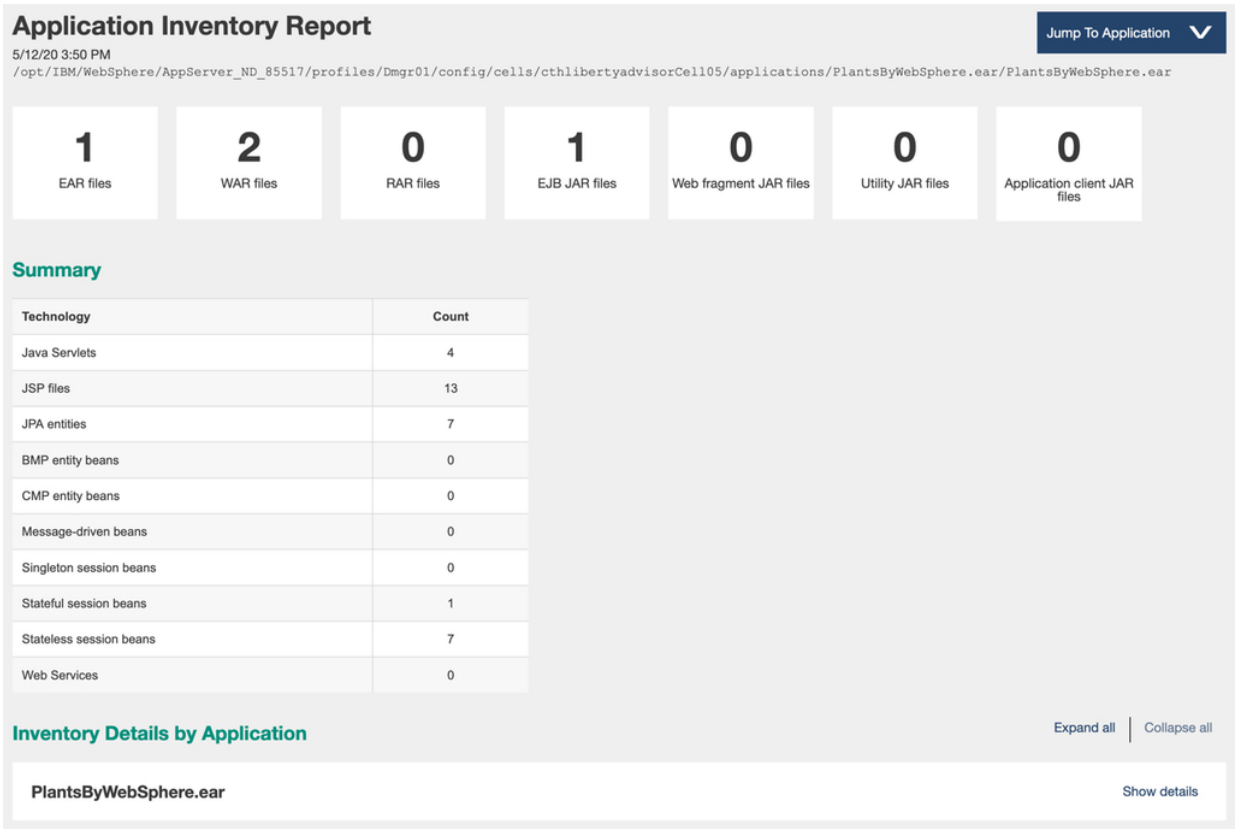
The highlighted columns indicate which IBM platforms fully support the technologies used by the included application.

Recommendation: Detailed migration analysis should be used to determine if there are migration issues that must be addressed before deploying your application.

	Liberty for Java on IBM Cloud	Liberty Core	Liberty	WebSphere traditional	Network Deployment Liberty	Network Deployment traditional	Liberty for z/OS	WebSphere traditional for z/OS
WEB SERVICES TECHNOLOGIES								
Java API for XML-based RPC (JAX-RPC)				✓		✓		✓
WEB APPLICATION TECHNOLOGIES								
Java Servlet	✓	✓	✓	✓	✓	✓	✓	✓
JavaServer Pages / Expression Language (JSP/EL)	✓	✓	✓	✓	✓	✓	✓	✓
ENTERPRISE APPLICATION TECHNOLOGIES								
Enterprise JavaBeans (EJB) 2.x and 1.x	✓		✓	✓	✓	✓	✓	✓
Enterprise JavaBeans (EJB) Lite subset	✓	✓	✓	✓	✓	✓	✓	✓
Remote Enterprise JavaBeans (EJB)			✓	✓	✓	✓	✓	✓
Java Persistence (JPA)	✓	✓	✓	✓	✓	✓	✓	✓
Common Annotations for the Java Platform	✓	✓	✓	✓	✓	✓	✓	✓
JavaMail	✓	✓	✓	✓	✓	✓	✓	✓
JAVA EE-RELATED SPECIFICATIONS IN JAVA SE								
Java Database Connectivity (JDBC)	✓	✓	✓	✓	✓	✓	✓	✓

Application inventory report

The Inventory Report helps you understand your application contents including the number of modules and the Java EE technologies used by those modules. It also gives you a view of all the utility JAR files in the application that tend to accumulate over time. Potential deployment problems and performance considerations are also included.



Application Inventory Report

Jump To Application

5/12/20 3:50 PM

/opt/IBM/WebSphere/AppServer_ND_85517/profiles/Dmgr01/config/cells/cthlibertyadvisorCell05/applications/PlantsByWebSphere.ear/PlantsByWebSphere.ear

1	2	0	1	0	0	0
EAR files	WAR files	RAR files	EJB JAR files	Web fragment JAR files	Utility JAR files	Application client JAR files

Summary

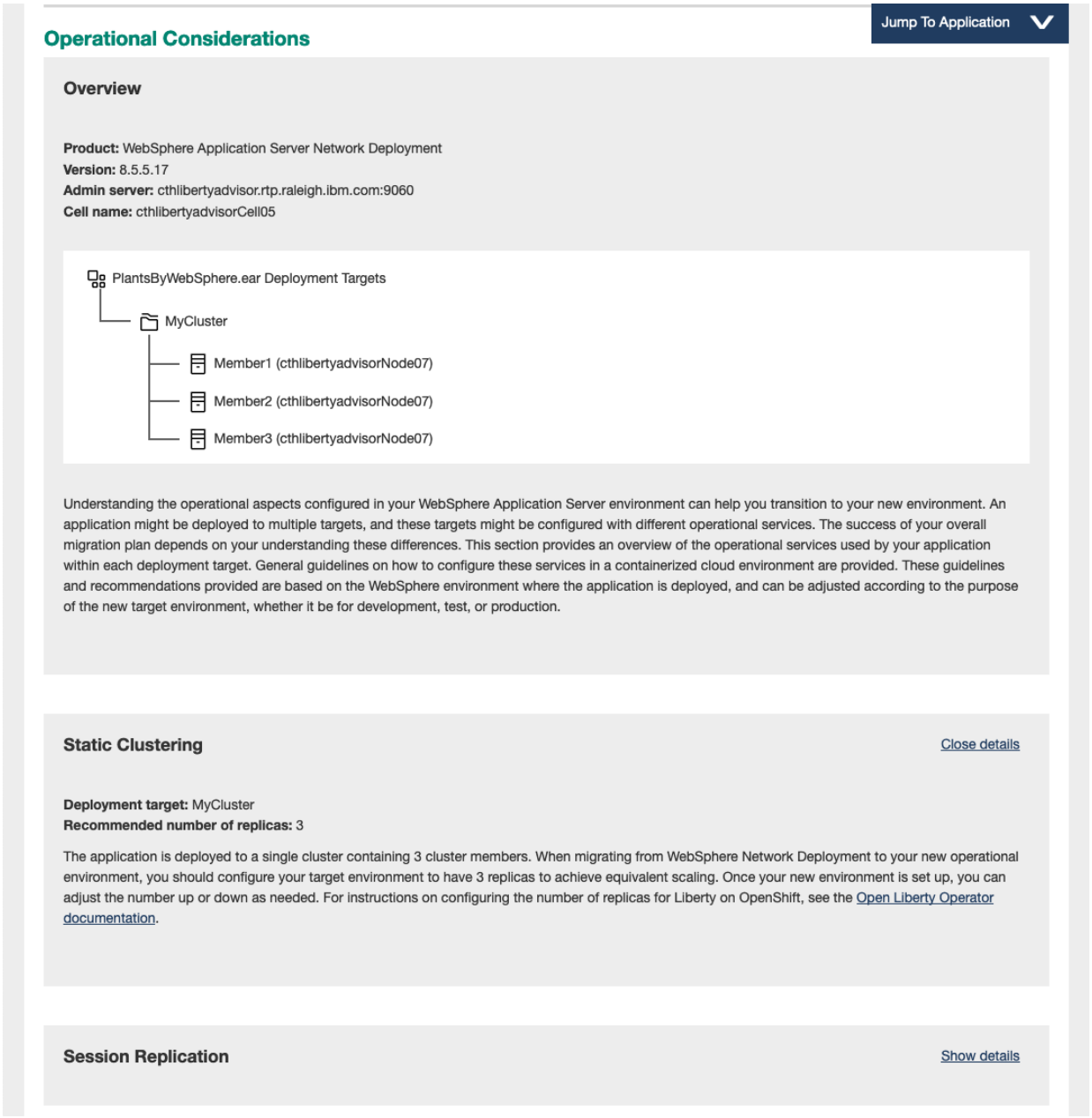
Technology	Count
Java Servlets	4
JSP files	13
JPA entities	7
BMP entity beans	0
CMP entity beans	0
Message-driven beans	0
Singleton session beans	0
Stateful session beans	1
Stateless session beans	7
Web Services	0

Inventory Details by Application

Expand all | Collapse all

PlantsByWebSphere.ear Show details

As of version 20.0.0.2, the Inventory Report includes new insights on how to migrate WebSphere Application Server Network Deployment qualities of service to a Kubernetes environment.

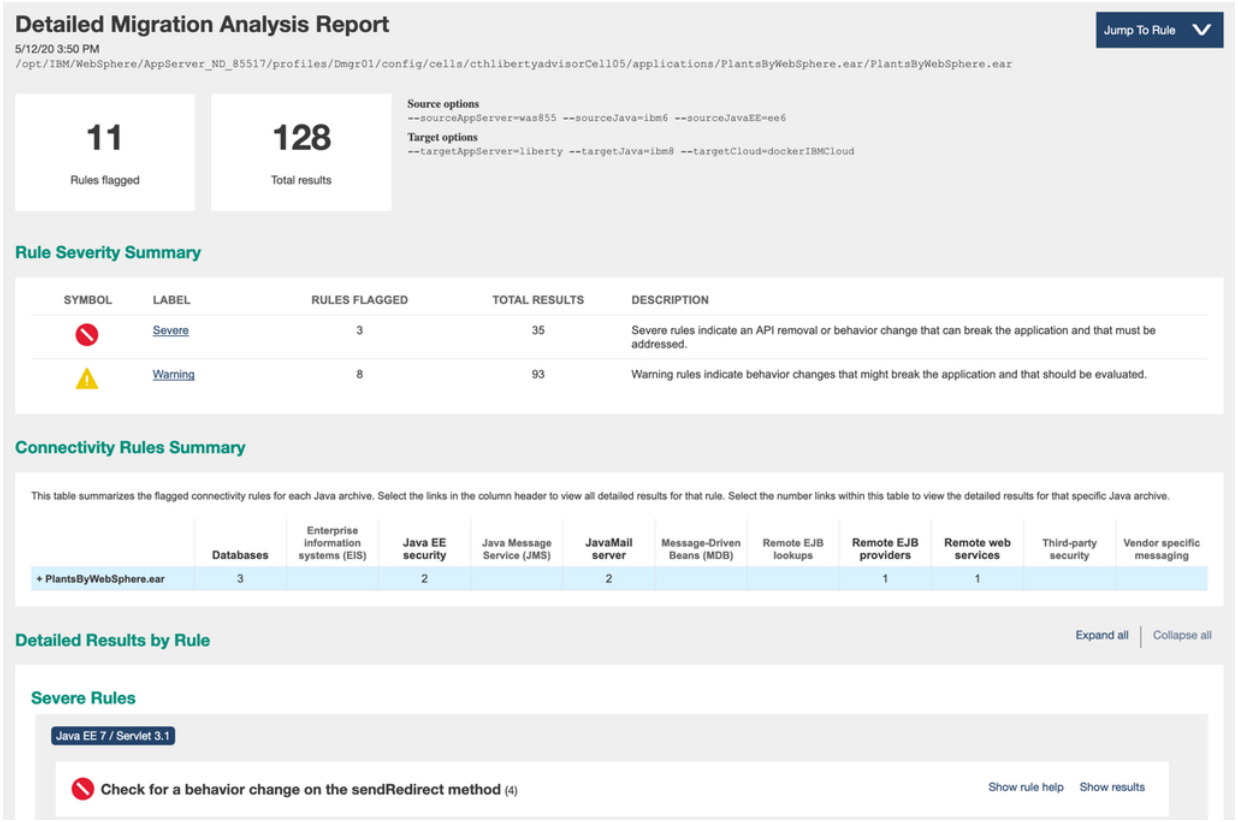


Detailed Migration Analysis Report

The Migration Analysis Report gives insights to dive in deeper to understand the details of the migration effort. Based on your source and target application server, Java EE levels, and Java SE levels, a set of analysis rules are run against your application binary files. The detailed analysis results give you insight to application issues such as those in the following list.

- Java EE differences going back as far as Java EE 6
- Changes to the Java Runtime Environment (JRE) encountered in going back as far as Java SE 5
- Removal of previously deprecated features
- Behavior changes in product APIs
- Changes resulting from Java EE specification clarifications
- Deprecated features
- WebSphere APIs not available on Liberty
- Optional Java EE technologies not available on Liberty
- Differences in technology implementations
- Cloud connectivity considerations
- Deployment descriptor differences for third-party application servers.

Every issue flagged has a detailed help with useful links to help you mitigate migration issues.



Liberty or WebSphere Application Server traditional configuration

When the application is scanned from a deployed environment or from backup configuration, the tools also produce configuration for deploying to Liberty or WebSphere Application Server traditional containerized environments. At a minimum for Liberty, a feature list is created which provides you the set of features needed to run the application. By adding only the features you need, you create a server that is right-sized for the application needs.

server.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!--Generated by IBM TransformationAdvisor
Tue May 12 23:54:04 UTC 2020--><server description="Configuration generated by binaryAppScanner">
    <featureManager>
        <!--The following features are available in all editions of Liberty.-->
        <feature>appSecurity-2.0</feature>
        <feature>ejbLite-3.2</feature>
        <feature>javaMail-1.5</feature>
        <feature>jdbc-4.1</feature>
        <feature>jndi-1.0</feature>
        <feature>jpa-2.0</feature>
        <feature>jsp-2.3</feature>
        <feature>servlet-3.1</feature>
        <!--The following features are available in all editions of Liberty, except for Liberty Core.-->
        <feature>ejbRemote-3.2</feature>
    </featureManager>
    <!-- This configuration was migrated on 2020-05-12 at 19:19:12 from the following location: /opt/IBM/WebSphere/AppServer_ND_
    <!-- The binary scanner does not support the migration of all WebSphere traditional configuration elements. Check the binary
    <applicationManager autoExpand="true"/>
    <httpEndpoint host="*" httpPort="9080" httpsPort="9443" id="defaultHttpEndpoint"/>
    <enterpriseApplication location="PlantsByWebSphere.ear"/>
<application id="PlantsByWebSphere" location="plantsbywebsphere-1.0.0.war" name="PlantsByWebSphere" type="war"/>
</server>
```

Transformation Advisor

The binary scanner can be used to create a Transformation Advisor data collection archive. You can manually upload the data collection into your running Transformation Advisor installation. Using the binary scanner gives you a jump on data collection when you do not have Transformation Advisor

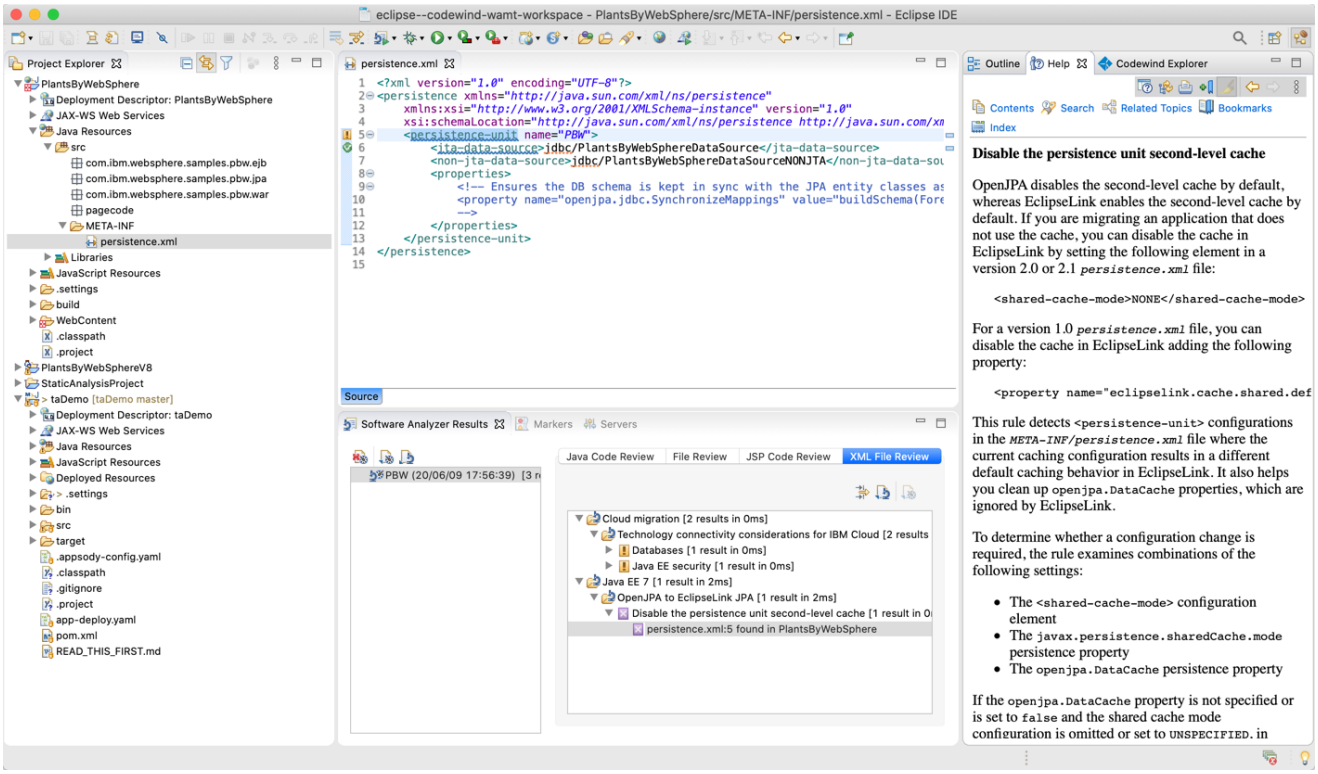
already installed. Use the **`java -jar binaryAppScanner.jar --help --ta`** command to get started with Transformation Advisor data collection.

WebSphere Application Server Migration Toolkit (source scanner)

After you complete the initial analysis by using Transformation Advisor, the admin console, or the binary scanner, the WebSphere Application Server Migration Toolkit Eclipse plug-in helps developers make the source code changes easier. You run the same set of rules based on your source and target application server, Java EE levels, and Java SE levels against your application source files including Java, JSP, XML, XMI, and properties files. Where possible, the Eclipse-based tool has quick fixes that you can optionally use to make changes to your code. When a quick fix is provided, you can use a side-by-side compare tool to see and understand the changes being recommended before you apply the fix.

Configure the analysis tool by selecting the Run > Analysis menu. Create a Software Analyzer configuration. Choose the scope of the analysis and select the rules to run. Select the *WebSphere Application Server Version Migration* **`Rule Set`** and click Set to configure the migration rules for your scenario.

The Software Analyzer Results view show the issues found in your code. From the result, you can double-click to open the source code in the editor and view the rule help in the Eclipse Help view.



Mono2Micro

When you want to do more than get your monolith application running on the most modern runtime, IBM Mono2Micro helps you analyze your application gain insights on how to break your monolith into microservices. IBM Mono2Micro uses machine learning to analyze your Java application at the class level based on runtime calls and detected data dependencies (specifically, containment and inheritance relationships). The analysis produces application refactoring options that can be explored and modified

through graphs and reports. Mono2Micro has code generation capability to help the initial steps of breaking the monolith apart.

WebSphere Liberty JAX-RPC Conversion Tool for Maven and Gradle

Since Liberty does not support JAX-RPC services or clients, the WebSphere Liberty JAX-RPC Conversion Tool is available to convert JAX-RPC applications to JAX-WS applications that can run on Liberty. This tool runs in Maven or Gradle as part of your build process. There is a validation task and a conversion task. As part of the binary scanner and Transformation Advisor data collection, applications are validated to see whether they are candidates to use this tool. See your analysis reports or Transformation Advisor output to understand whether your applications can take advantage of this tool.

Eclipse Transformer

The Eclipse Transformer helps you modernize applications to use the Jakarta EE programming model. Before Jakarta 9, the Java EE programming model uses **`javax`** in their package naming. Starting with Jakarta 9, the Jakarta programming model moved to **`jakarta`** package naming. The Eclipse Transformer helps you modify either your application source code or your binary archives to use the new package naming. It takes the guess work out of the package rename since some Java SE classes still use **`javax`**.

Migrating WebSphere Application Server traditional versions

Move your applications to a lightweight application server for a cloud environment or if you want to take advantage of newer Java™ SE, Java EE, or Jakarta EE capabilities. However, if there is a reason to perform WebSphere Application Server traditional release to release migrations, refer to the following information.

WebSphere Application Server Product Service Announcements

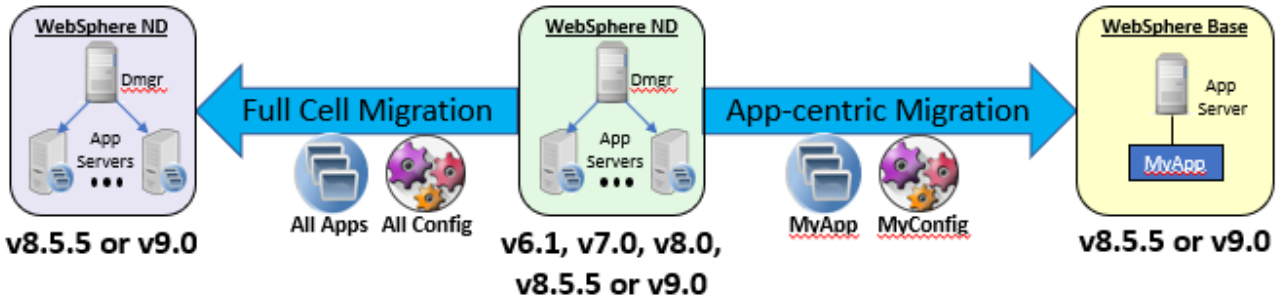
Before you proceed with the migration, review the following traditional WebSphere support announcements.

- WebSphere Application Server Support Restatement
- Revised support for WebSphere Application Server V8.5.5 and V9.0.5 (Announcement Letter: 220-128) - If you are already running on WebSphere Application Server V8.5.5, you get the same support duration as V9 without migrating. See what's new in the latest V9 release, but before you start any WebSphere tradition migration, take a look at Liberty as your migration target.
- End of Support for WebSphere Application Server V7 and V8 (Announcement Letter: 916-143)
- End of Support for z/OS WebSphere Application Server V7 and V8 (Announcement Letter: 916-159)

Migration overview

There are many options available when you perform a WebSphere Application Server traditional release to release migration. You can migrate your entire WebSphere Application Server Network Deployment (ND) cell into a later release of the WebSphere Application Server product. You can migrate a single application from

your ND cell into its own containerized WebSphere Application Server base environment, which can then be deployed to the cloud. Continue reading to learn more about the processes and tools necessary to carry out either a full cell migration, an application-centric cloud migration, or a z/OS migration.



Although different tools are used to perform a distributed WebSphere cell versus z/OS versus app-centric migration, the process is similar. The migration process can be divided into three steps:

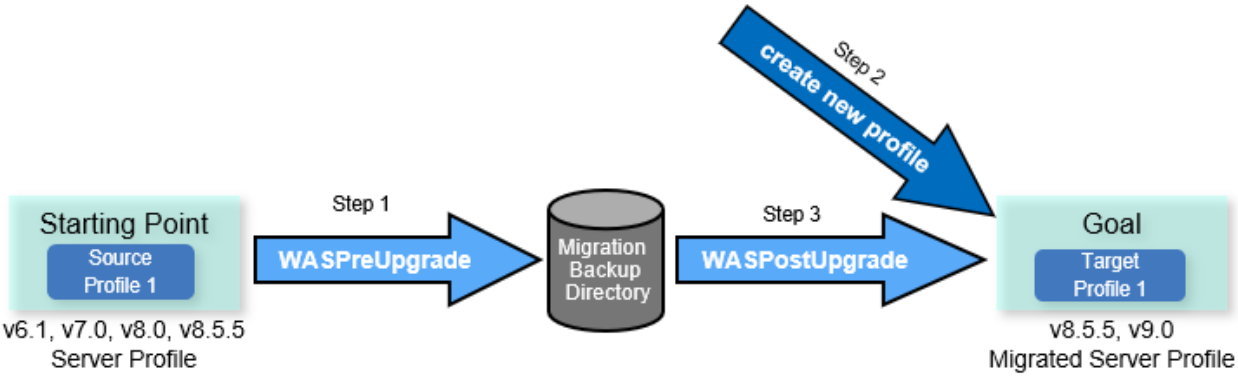
1. Capture the applications and the associated configuration.
2. Create a target profile in the new WebSphere Application Server release.
3. Apply the captured configuration to the target profile and deploy the applications.

The intention of this process is that the new environment functions as closely as possible to the previous one. At the same time, it gives you the ability to upgrade the configuration to use the new features and functions in the new release.

Full-cell distributed migration

Each WebSphere Application Server release is shipped with an updated set of migration tools that must be used to upgrade to the new release. These tools work on a profile-by-profile basis, whereby the following process is repeated for each profile in the cell, starting with the deployment manager:

1. **`WASPreUpgrade`**: captures the old configuration data and applications.
2. **`manageprofiles`**: creates a target profile in the new release.
3. **`WASPostUpgrade`**: merges in the old configuration data into the target profile and installs the applications.



The full cell migration process supports the following and various other options:

- Migrating to new host machines.
- Cloning your active cell to v9.0 while your current cell stays functional.
- Selecting only those applications you want moved forward.

Using the machine change, clone and selective application options together provide you with the ability to split the cell into multiple cells, where you can then deploy a specific set of applications into each clone. You

can leave non-strategic applications running in their current environment while you move your mission-critical applications to the new release.

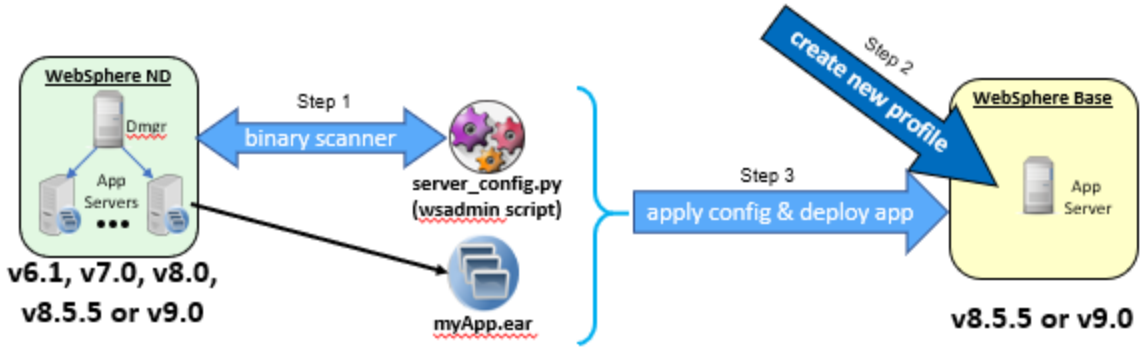
App-centric migration for cloud deployments

Liberty is the ideal containerized runtime, as discussed in Migrating to Liberty. However, sometimes there are valid reasons for not modernizing applications to Liberty. For applications that you plan to decommission soon or that do not easily migrate to Liberty, you can containerize your application in a WebSphere Application Server base server so that all applications are managed with the same operational model. Today, you might host many applications on a single traditional WebSphere Application Server. When you deploy containerized applications to a cloud environment, it is considered best practice to limit the deployment to a single application or as few applications as possible. This practice makes container image maintenance, updates, and scaling easier.

IBM Cloud Transformation Advisor supports migrating a single application from a WebSphere Application Server environment by generating a configuration script that can be used to configure the server. In addition to providing configuration, IBM Cloud Transformation Advisor creates other deployment artifacts, such as a Dockerfile and configuration yaml files, that are used for deploying WebSphere Application Server traditional images to Kubernetes. IBM Cloud Transformation Advisor is included with IBM WebSphere Hybrid Edition and has a free local trial version.

You can also use the WebSphere Application Server Migration Toolkit for Application Binaries (binary scanner) to generate an application-specific WebSphere Application Server traditional configuration script. Use the **`--generateConfig`** and **`--targetServer=was90`** or **`was855`** options in the binary scanner to produce a **`wsadmin`** script that can configure your new target profile to support the specified application that you are installing.

You can learn more about the WebSphere Application Server traditional container images that IBM publishes and the best practices for their use.



z/OS migration

The tools and process for a WebSphere z/OS migration are different from the ones used on distributed systems, but the underlying concepts are the same. Although z/OS does not support remote migrations, it does support the cloning process. Review the following z/OS V9.0 IBM documentation migration links to get started:

- WebSphere Customization Toolbox installation information (WCT)
- z/OS Migration Considerations
- Preparing to migrate a WebSphere Application Server Network Deployment cell for z/OS
- Using the z/OS Migration Management Tool to create and manage migration definitions (zMMT)

- Migrating, coexisting, and interoperating

Migration planning

You are encouraged to always migrate to the latest fix pack of your migration target so that you have the latest migration tools. Start your planning by reviewing and becoming familiar with the information provided in the following documents:

- Traditional WebSphere Migration Overview - IBM Documentation
- WebSphere Application Server Migration Planning and Best Practices Guide (6 MB)
- WebSphere Application Server Versions What's Different (4 MB)
- Handling Application Deployment Issues During Migration (120 KB)

Migration tool reference and links

The following list of the migration tools provides a brief description of the tool and a reference link for more information as you plan your WebSphere Application Server migration.

Tools included in WebSphere Application Server

- **createRemoteMigrJar**: facilitates cross machine migrations by eliminating the need to install the new release of WebSphere Application Server on your old machine just to run the WASPreUpgrade command.
- **WASPreUpgrade**: captures a snapshot of a WebSphere Application Server profile's configuration data and installed applications and saves it to a migration backup directory.
- **WASPostUpgrade**: merges the configuration data captured by the WASPreUpgrade command into the target profile created in the new WebSphere Application Server release.
- **WASMigrationAppInstaller** (v9 only): installs the applications captured by the WASPreUpgrade command into the target profile created in the new WebSphere Application Server release. Use this command when the `-includeApp script` option was used with the WASPostUpgrade command. Select the applications you want to move forward with this command.

Wizards

- **WebSphere Application Server Migration Wizard**: guides you through the migration process on the distributed platforms. Runs the WASPreUpgrade, profile creation, and WASPostUpgrade commands. This tool can also be used to generate the commands that are needed to run the migration from the command line.
- **WebSphere Customization Toolbox** (WCT): suite of tools that create profiles and migrate them on the z/OS platform.

Other migration tools

- **IBM WebSphere Migration Toolkit for Application Binaries** (binary scanner): used to analyze an application in its current WebSphere environment. Produces reports to help assess compatibility with your target environment. Can be used to perform app-centric migrations.
- **WebSphere Application Server Migration Toolkit** (source scanner): an Eclipse IDE developers' tool that helps pinpoint and flag changes to the application source files so they can run in the new server environment. The flagged changes include WebSphere APIs, Java SE, and Java EE level differences.
- **IBM Cloud Transformation Advisor**: provides a detail analysis of your application suite deployed in your WebSphere Application Server environment. Provides recommendations,

guidance, and associated artifacts needed to migrate your applications to their new server environment.

Downloads

Access links to downloads for WebSphere Application Server Migration Toolkit and IBM Cloud Transformation Advisor.

Table 1. Downloads for WebSphere Application Server Migration Toolkit, IBM Cloud Transformation Advisor, and IBM Mono2Micro

Download	Description
Migration Toolkit for Application Binaries (September 2023)	The Migration Toolkit for Application Binaries provides a command-line tool that quickly evaluates application code and configuration for rapid deployment on Liberty or newer versions of WebSphere Application Server traditional.
WebSphere Application Server Migration Toolkit (September 2023)	The migration toolkit provides Eclipse-based tools for application source migration scenarios that include Liberty migration, cloud migration, WebSphere version-to-version migration, and migration from third-party application servers.
IBM Cloud Transformation Advisor (September 2023)	IBM® Cloud Transformation Advisor™ analyzes your on-premises workloads for modernization to the Cloud. It determines the complexity of your applications, estimates the development cost to move to the Cloud, and provides recommendations for the best target environment.
IBM Mono2Micro (October 2023)	IBM Mono2Micro uses AI to help you transform your monolithic applications into microservices; it offers suggestions on microservice partitions to consider based on business use cases, and it also generates microservice code that you can use as a starting point to speed up your refactoring efforts.

Additional resources

Use these links to documents and resources to help you plan and perform your WebSphere Application Server migrations and modernizations.

General Information

- Learning Path: IBM WebSphere Hybrid Edition
- IBM Skills Gateway - WebSphere Application Server
- IBM Software Product Compatibility Reports
- IBM WebSphere, Liberty Java™ & DevOps Community
- My Notifications subscription for critical IBM software support updates

Modernization and Migration Tools

- IBM Cloud Transformation Advisor
- Generate migration reports with the traditional WebSphere Admin Console
- Migration Toolkit for Application Binaries download (binary scanner)
- WebSphere Application Server Migration Toolkit download (source code scanner)
- createRemoteMigrJar
- WASPreUpgrade
- WASPostUpgrade
- WASMigrationAppInstaller(v9 only)
- WebSphere Migration Wizard
- WebSphere Customization Toolbox
- Migrating JAX-RPC applications to Liberty

WebSphere Application Server traditional to Liberty

- Open Liberty
- Open Liberty Guides
- WebSphere Application Server Liberty documentation

WebSphere Application Server traditional migration

- Migrating, coexisting, and interoperating
- Migrating cells to new host machines by using the command-line tool
- Deprecated, stabilized, and removed features of WebSphere Application Server traditional
- Default value and behavior changes from previous releases of WebSphere Application Server traditional
- Redbooks: WebSphere Application Server V8.5 Migration Guide
- Redbooks: WebSphere Application Server V8.5 Concepts, Planning, and Design Guide
- Redbooks: WebSphere Application Server V8.5.5 Technical Overview
- Redbooks: WebSphere Application Server: New Features in V8.5.5
- Best Practices for Integrating Open Source Software Frameworks with WebSphere Application Server

Java SE Migration

- Java SE 7 and JDK 7 Compatibility Guide
- IBM SDK, Java Technology Edition, Version 8
- Compatibility Guide for JDK 8
- Java SE 11 - Oracle JDK Migration Guide
- Java SE 17 - Oracle JDK Migration Guide
- Semeru Runtimes migration guide

Java EE Migration

- WebSphere traditional programming model APIs and specifications
- Liberty Java EE 7 programming model documentation
- Liberty Java EE 8 programming model documentation
- Liberty Jakarta EE 9.1 programming model documentation
- How To Migrate from OpenJPA to EclipseLink JPA
- Migrating WebSphere Technologies

- WebSphere Liberty JAX-RPC Conversion Tool for Maven and Gradle
- Eclipse Transformer for Jakarta EE migration

Support and Services

- IBM Support
- IBM Software Accelerated Value Program
- Migration Assist - WebSphere Application Server Level 2 Support team provides support to clients planning a release to v8.5 or v9.0 migration.
- No-charge Migration Waiver - Facilitate your migration by getting temporary Subscription & Support (S&S) rights to use your current licenses in production and concurrently use the later version during the migration period subject to certain requirements. Contact your IBM representative for more details.

Frequently asked questions

Get answers to frequently asked questions about moving applications to the cloud.

- What is the official Maven groupId?
- Where are WebSphere Liberty images published?
- What tagging mechanism is used for the images?
- How do I understand the choices of base image (UBI, Ubuntu)?
- What is the support policy?
- How do I find the Open Liberty operator?
- What tools can I use to develop my cloud-native application?
- What do I use to build a Liberty image?
- What OpenShift versions are supported?
- Where do I get trial versions of the IBM WebSphere Hybrid Edition components?

What is the official Maven groupId?

See the following sites:

- https://search.maven.org/
- openliberty.io/docs
- https://central.sonatype.com/namespace/io.openliberty.features

Where are WebSphere Liberty images published?

See https://www.ibm.com/docs/en/was-liberty/nd?topic=images-liberty-container.

What tagging mechanism is used for the images?

See https://www.ibm.com/docs/en/was-liberty/nd?topic=images-liberty-container.

Only features that have been explicitly declared in the WebSphere Liberty server.xml file will be installed into the image

How do I understand the choices of base image (UBI, Ubuntu)?

See https://www.ibm.com/docs/en/was-liberty/nd?topic=images-liberty-container.

What is the support policy?

See https://www.ibm.com/docs/en/was-liberty/nd?topic=running-websphere-liberty-in-container.

How do I find the Open Liberty operator?

See https://www.ibm.com/docs/en/was-liberty/nd?topic=images-liberty-container.

What tools can I use to develop my cloud-native application?

- Eclipse IDE (https://www.eclipse.org/downloads/)
- WebSphere Liberty and Java™ SE 8 (https://www.ibm.com/support/pages/websphere-liberty-developers)
- WebSphere Liberty Plugin for Eclipse IDE installed with WAMT - WebSphere Application Server Migration Toolkit (https://www.ibm.com/support/pages/websphere-liberty-developers)
- Java JDK & Runtime 17 (https://developer.ibm.com/languages/java/semeru-runtimes/downloads/)
- Podman (https://podman.io)
- Access to OpenShift (installed OpenShift CLI) and to the public/private registry where to push the Application Image (edited)

What do I use to build a Liberty image?

See https://develop.cloudnativetoolkit.dev/reference/starter-kit/starter-kits/.

What OpenShift versions are supported?

See https://www.ibm.com/docs/en/was-liberty/nd?topic=requirements-openshift-container-platform.

Where do I get trial versions of the IBM WebSphere Hybrid Edition components?

See https://www.ibm.com/cloud/blog/websphere-trial-options-and-downloads.

Installing

When you purchase IBM WebSphere Hybrid Edition, you get the software available with WebSphere Hybrid Edition in one bundle. You can choose to deploy some or all the purchased software.

About this task

These installation instructions are for on-premises installation.

Installing the WebSphere Hybrid Edition on premises

To install WebSphere Hybrid Edition software on premises:

1. Purchase WebSphere Hybrid Edition.
2. Follow the installation instructions for the software that you want to install.

WebSphere Application Server Network Deployment
: Downloading and installation instructions for WebSphere Application Server Network Deployment and WebSphere Application Server Network Deployment Liberty are available in Installing WebSphere Application Server Network Deployment.

WebSphere Application Server
: Downloading and installation instructions for WebSphere Application Server and WebSphere Application Server Liberty are available in Installing WebSphere Application Server.

WebSphere Application Server Liberty Core
: Downloading and installation instructions for WebSphere Application Server Liberty Core are available in Installing WebSphere Application Server Liberty Core.

IBM Cloud Transformation Advisor
: Prerequisite information and installation instructions are available in Installing and upgrading IBM Cloud Transformation Advisor.

IBM Mono2Micro
: Prerequisite information and installation instructions are available in Installing IBM Mono2Micro.

WebSphere Application Server Migration Toolkit
: Downloading and installation instructions are available in Installing WebSphere Application Server Migration Toolkit.

Ordering an IBM WebSphere Hybrid Edition offering

To order IBM WebSphere Hybrid Edition, consult your IBM representative or authorized IBM Business Partner.

The following tables list the available IBM WebSphere Hybrid Edition virtual processor core (VPC) offerings.

IBM WebSphere Hybrid Edition offerings

Table 1. IBM WebSphere Hybrid Edition part number descriptions and part numbers

Part number description	Part number
IBM WebSphere Hybrid Edition Virtual Processor Core License + SW Subscription & Support 12 Months	D299MLL
IBM WebSphere Hybrid Edition Virtual Processor Core Annual SW Subscription & Support Renewal 12 Months	E0R7ULL
IBM WebSphere Hybrid Edition Virtual Processor Core SW Subscription & Support Reinstatement 12 Months	D299NLL
IBM WebSphere Hybrid Edition Virtual Processor Core from Prior Programs Trade up License + SW Subscription & Support 12 Months	D299PLL
IBM WebSphere Hybrid Edition Virtual Processor Core Monthly License	D299YLL

Part number description	Part number
IBM WebSphere Hybrid Edition Virtual Processor Core Committed Term License	D29A0LL
IBM WebSphere Hybrid Edition Virtual Processor Core Committed Term License Upgrade	D29A2LL

What to do next

After you order IBM WebSphere Hybrid Edition, see the installation instructions to install the software.

Software Product Compatibility Reports (SPCR)

When you plan an installation, review the list of supported operating systems, detailed system requirements, and hardware requirements. To get this information, you can view a software product compatibility report.

1. Go to the Software Product Compatibility Reports page.
2. Click Create a report for the type of report you want to create.
3. For the Full or partial product name field, enter `WebSphere Hybrid Edition` and click the Search product icon.
4. In the search results, select a version in the Version list and click Submit.

 Watch the Using IBM's Software Product Compatibility Reports video for a demonstration of how to generate custom reports about compatible IBM software combinations.

Installing WebSphere Application Server Network Deployment

Learn about prerequisites and instructions for downloading and installing WebSphere Application Server Network Deployment.

Downloading WebSphere Application Server Network Deployment

To download WebSphere Application Server Network Deployment, see Passport Advantage.

Installing WebSphere Application Server Network Deployment

To install WebSphere Application Server Network Deployment, see the WebSphere Application Server Network Deployment documentation.

Installing WebSphere Application Server Network Deployment Liberty

To install WebSphere Application Server Network Deployment, see the WebSphere Application Server Network Deployment documentation.

Installing WebSphere Application Server Liberty Core

Learn about prerequisites and instructions for downloading and installing WebSphere Application Server Liberty Core.

Downloading WebSphere Application Server Liberty Core

To download WebSphere Application Server Liberty Core, see Passport Advantage.

Installing WebSphere Application Server Liberty Core

To install WebSphere Application Server Liberty Core, see the WebSphere Application Server Liberty Core documentation.

Installing WebSphere Application Server

Learn about prerequisites and instructions for downloading and installing WebSphere Application Server.

Downloading WebSphere Application Server

To download WebSphere Application Server, see Passport Advantage.

Installing WebSphere Application Server

To install WebSphere Application Server, see the WebSphere Application Server documentation.

Installing WebSphere Application Server Liberty

To install WebSphere Application Server, see the WebSphere Application Server documentation.

Installing and upgrading IBM Cloud Transformation Advisor

Learn about prerequisites and instructions for downloading, installing, and upgrading IBM Cloud Transformation Advisor.

Installing IBM Cloud Transformation Advisor

To install the latest version of IBM Cloud Transformation Advisor, see the IBM Cloud Transformation Advisor documentation.

Upgrading IBM Cloud Transformation Advisor

To upgrade to the latest version of IBM Cloud Transformation Advisor, see the IBM Cloud Transformation Advisor documentation.

Installing IBM Mono2Micro

Learn about prerequisites and instructions for downloading and installing IBM Mono2Micro.

For prerequisites, instructions on obtaining and using an entitlement key, instructions for downloading assets and Docker images, and instructions for installing IBM Mono2Micro, see the IBM Mono2Micro documentation.

Installing WebSphere Application Server Migration Toolkit

Learn about prerequisites and instructions for downloading and installing WebSphere Application Server Migration Toolkit.

For prerequisites and instructions for downloading and installing WebSphere Application Server Migration Toolkit, see the WebSphere Application Server Migration Toolkit documentation.

Installing and updating the WebSphere Liberty operator

Learn about prerequisites and instructions for downloading, installing, and upgrading the WebSphere Liberty operator.

Installing WebSphere Liberty operator

To install the WebSphere Liberty operator, see the WebSphere Liberty documentation.

Updating WebSphere Liberty

To update the WebSphere Liberty operator, see the WebSphere Liberty documentation.

Setting up a cluster for an air gap installation

Follow the instructions to set up air gap installation for the included components.

- Installing IBM Cloud Transformation Advisor in an air gap environment
- For Open Liberty operator and Runtime Component operator, see Using Operator Lifecycle Manager on restricted networks

Applying interim fixes to runtimes in containers

It is often necessary to apply interim fixes (ifixes) to your runtime container images. Instructions for applying ifixes are available at the indicated locations.

Applying interim fixes to Liberty in a container

For information on applying interim fixes to an instance of Liberty that is running in a container, see Applying interim fixes page on the Liberty container images wiki

Applying interim fixes to WebSphere Application Server in a container

For information on applying interim fixes to an instance of WebSphere Application Server that is running in a container, see Installing ifixes section on the WebSphere traditional container images page

WebSphere Application Server editions

WebSphere Application Server, with its traditional and Liberty runtimes, offers production-ready, standards-based compliance to support the Application Modernization strategies that underpin business transformation. Additional features and enhancements to WebSphere Application Server offer an ideal infrastructure that is well-suited for enterprise IT, upon which businesses can deliver composable applications and enhancements to help operational modernization.

Converging the operational models of traditional WebSphere Application Server and Kubernetes takes advantage of Kubernetes platform services, such as logging and monitoring. This helps to enable secure, flexible, and efficient access to internal or external software components and services. These enhancements ease integration of WebSphere runtimes in the DevOps workflows to provide continuous integration and continuous delivery to container-based Kubernetes environments such as IBM Cloud.

WebSphere Hybrid Edition supports on-premises, cloud, and hybrid cloud deployments and includes the following WebSphere Application Server editions:

- WebSphere Application Server Network Deployment
- WebSphere Application Server Liberty Core
- WebSphere Application Server

WebSphere Application Server Network Deployment

WebSphere Application Server Network Deployment edition includes the Network Deployment editions of both the traditional WebSphere Application Server and Liberty application servers.

WebSphere Application Server Network Deployment edition offers enterprises near-continuous availability, advanced management, and automated performance optimization for their mission-critical applications. WebSphere Application Server Network Deployment capabilities include a superset of those found in WebSphere Application Server Liberty Core and the single-server WebSphere Application Server configuration.

For more information, see the documentation:

- WebSphere Application Server Network Deployment
- WebSphere Application Server Network Deployment - Liberty option

WebSphere Application Server

WebSphere Application Server includes the single server editions of both the traditional WebSphere Application Server and Liberty application servers.

WebSphere Application Server is tailored for a single server to moderately-sized configurations of departmental or large-scale, dynamic web applications that require web-tier clustering and failover across application server instances. The Liberty capabilities include a superset of those found in WebSphere Application Server Liberty Core.

For more information, see the documentation:

- WebSphere Application Server
- WebSphere Application Server Liberty

WebSphere Application Server Liberty Core

WebSphere Application Server Liberty Core, a lightweight and dynamic offering of WebSphere Application Server, is Java™ EE 8 web profile compliant. Composed entirely of the developer-friendly Liberty profile, the WebSphere Application Server Liberty Core edition enables rapid development and deployment of web-centric and mobile-centric applications.

WebSphere Application Server Liberty Core allows businesses to quickly respond to enterprise and market needs. Liberty profile servers can be members of collectives that enables them to be managed by a collective controller from a Network Deployment installation. The capabilities that are provided in the Liberty Core edition are a subset of the capabilities that are provided in the base edition of WebSphere Application Server and the WebSphere Application Server Network Deployment edition.

Highlights of the Liberty Core offering include:

- Extremely lightweight offering that is composed of a subset of the Liberty profile, which is certified for the Java EE 8 Web Profile and implements the latest MicroProfile specifications in addition to Java EE 8
- An excellent development and production runtime for web applications
- Fast time-to-value
- Small download, small footprint, fast startup, and easily packaged applications. Includes configuration for deployment and is extensible through WebSphere Liberty features SPI

For more information, see the documentation:

- WebSphere Application Server Liberty Core

Application modernization tools

WebSphere Hybrid Edition includes the following options and features to enhance your WebSphere cloud experience.

- IBM Cloud Transformation Advisor: Helps businesses modernize and migrate their applications from on-premises environments to the cloud or to containers. Transformation Advisor can identify applications that are good candidates for migration and provide advice and recommendations regarding how to migrate that application.
- IBM Mono2Micro: An AI-driven feature based on IBM Research technology that accelerates and can take the risk out of refactoring existing applications into modern microservices that are ready for cloud deployments.
- WebSphere Application Server Migration Toolkit: A set of Eclipse-based tools for WebSphere migration scenarios including cloud migration, WebSphere version to version migration including WebSphere Liberty, and migration from third-party application servers.

IBM Cloud Transformation Advisor

IBM Cloud Transformation Advisor helps you plan, prioritize, and package your on-premises workloads for modernization on WebSphere software.

IBM Cloud Transformation Advisor gathers preferences about your on-premises and wanted cloud environments and then analyzes existing middleware deployments by using a data collector. After you upload the results of the data collector, you can review recommendations for migrating your applications to different cloud platforms and the estimated effort to migrate and modernize. IBM Cloud Transformation Advisor also creates necessary deployment artifacts to accelerate your migration to the cloud.

To learn more about IBM Cloud Transformation Advisor, see the IBM Cloud Transformation Advisor documentation.

IBM Mono2Micro

IBM Mono2Micro uses machine learning to analyze your Java™ application at the class level based on runtime calls and detected data dependencies (specifically, containment and inheritance relationships).

The analysis produces two alternative refactoring options for your application which can be explored in graphs and reports. Once you determine which refactoring option will be best for you, IBM Mono2Micro automatically produces code to deploy your application on Liberty as microservices with minimal rewriting necessary.

For instructions on installing and getting started with IBM Mono2Micro, see Installing IBM Mono2Micro. If you already have IBM Mono2Micro installed, view the product documentation.

WebSphere Application Server Migration Toolkit

WebSphere Application Server Migration Toolkit provides tools for WebSphere migration scenarios including cloud migration, WebSphere version to version migration including Liberty, migration from third-party application servers, and Java™ SE migrations.

The WebSphere Application Server Migration Toolkit is an Eclipse plugin that analyzes application source code. The tool gives insights to help you understand the details of your migration effort. Based on your source and target application server, Java EE levels, and Java SE levels, a set of rules are run against your application source. Every issue flagged has detailed help with useful links to help you mitigate migration issues. Where possible, the Eclipse-based tool has quick fixes that you can optionally use to make changes to your code. When a quick fix is provided, you can use a side-by-side compare tool to see and understand the changes being recommended before you apply the fix. To get started with WebSphere Application Server Migration Toolkit, see the WebSphere Application Server Migration Toolkit documentation.

The Migration Toolkit for Application Binaries provides a command line tool that quickly evaluates application binaries. The tool analyzes the same analysis rules as the WebSphere Application Server Migration Toolkit Eclipse plugin based on your source and target application server, Java EE levels, and Java SE levels. The Migration Toolkit for Application Binaries also produces an inventory report to help you understand your application contents, as well as a technology report to help you understand the editions of WebSphere Application Server that are best suited to run the application. When the application is scanned from a deployed WebSphere Application Server traditional environment or backup configuration, the tool also produces a configuration for deploying to Liberty or traditional WebSphere containerized environments. To get started with Migration Toolkit for Application Binaries, see the Migration Toolkit for Application Binaries documentation.

Securing the environment

This information applies to all types of applications that are deployed on WebSphere Hybrid Edition.

- **Identity providers**
  Identity providers are web services that authenticate users on behalf of web applications. As an administrator, you can configure identity providers for web applications.
- **Open Liberty authentication**
  The Open Liberty operator in Red Hat OpenShift allows you to easily configure and manage the single sign-on with OpenID Connect or social medium for your applications.
- **Platform Application authentication**
  Your applications can be authenticated with Red Hat® OpenShift® native authentication service or IBM Cloud Pak foundational services Identity and Access Management service, and does not need provide its own authentication.
- **Certificate management and TLS**
  You need configure applications to provide secured communications between a client and the server with transport layer security (TLS). To enable TLS, X509 certificate is required.

Identity providers

Identity providers are web services that authenticate users on behalf of web applications. As an administrator, you can configure identity providers for web applications.

WebSphere Hybrid Edition allows administrators to easily configure and manage user authentication information for their applications through single sign-on with OAuth and OpenID Connect identity providers. WebSphere Hybrid Edition includes a Red Hat® Single Sign-On server (RH-SSO).

OAuth

OAuth is an open standard for access delegation, commonly used as a way for internet users to grant applications access to their information but without giving them the passwords. This mechanism is commonly used by popular social medium such as Google, Facebook, Github, and Twitter to permit the users to share information about their accounts with third party application.

OpenID Connect

OpenID Connect is a simple identity layer built on the OAuth 2.0 protocol, which allows applications to verify an end-user's identity based on the authentication performed by an authorization server, as well as to obtain basic profile information about the end-user in an interoperable and REST-like manner.

- **Red Hat Single Sign-On (RH-SSO)**
  Red Hat Single Sign-On (RH-SSO) is included with WebSphere Hybrid Edition; it is based on the Keycloak project and enables you to secure your web applications by providing web single sign-on (SSO) capabilities. The RH-SSO server can act as a SAML or OpenID Connect-based Identity Provider, mediating with your enterprise user directory or third-party SSO provider for identity information.
- **IBM Security Verify**
  You can configure IBM Security Verify as the identity and access management (IAM) solution for applications that are developed for WebSphere Hybrid Edition. IBM Security Verify is a managed, highly available identity as a service (IDaaS) provider that protects applications that are deployed across multiple clouds.

Red Hat Single Sign-On (RH-SSO)

Red Hat® Single Sign-On (RH-SSO) is included with WebSphere Hybrid Edition; it is based on the Keycloak project and enables you to secure your web applications by providing web single sign-on (SSO) capabilities. The RH-SSO server can act as a SAML or OpenID Connect-based Identity Provider, mediating with your enterprise user directory or third-party SSO provider for identity information.

IBM Security Verify

You can configure IBM Security Verify as the identity and access management (IAM) solution for applications that are developed for WebSphere Hybrid Edition. IBM Security Verify is a managed, highly available identity as a service (IDaaS) provider that protects applications that are deployed across multiple clouds.

Before you begin

The following resources are required before you can configure IBM Security Verify:

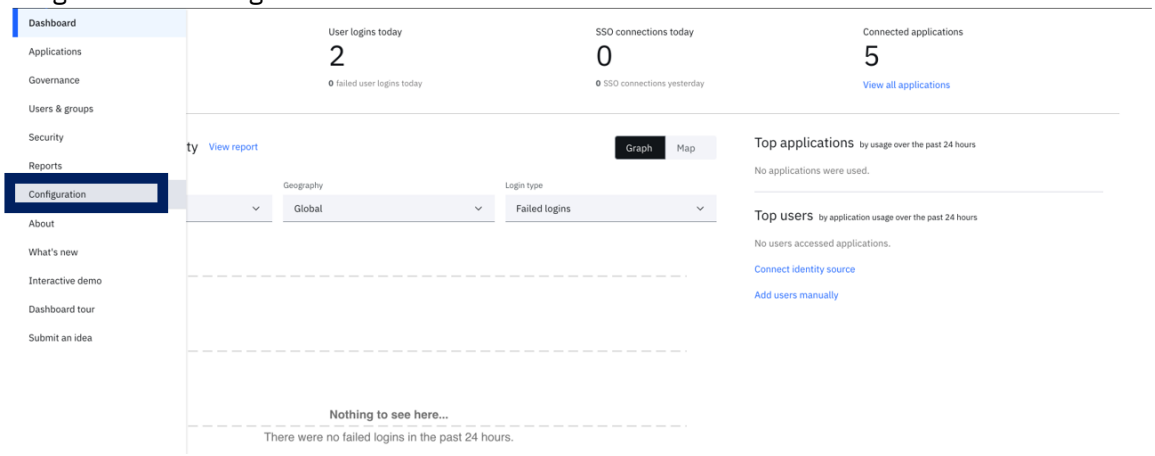
- A deployed WebSphere Hybrid Edition environment.
- An IBM Security Verify tenant. If you don't already have a tenant, you can create a free tenant.

Configuration steps

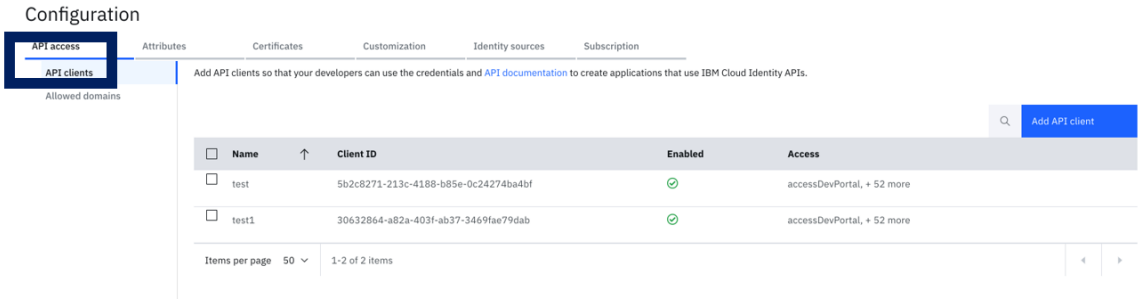
The configuration steps are divided into two sections. In the first section, an organization administrator sets up the OIDC IdP. In the second section, developers build application images with the OIDC details provided by the administrators.

For identity provider administrators:

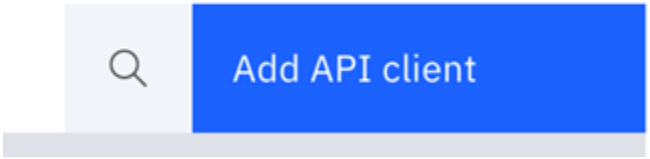
1. Log in to IBM Security Verify. If you cannot find your IBM Security Verify tenant URL, the welcome email from **`ibmacct@iam.ibm.com`** contains your tenant details. Your IBM Security Verify tenant URL is https://<tenant-id>.ice.ibmcloud.com/ui/admin.
2. Navigate to the Configuration section.



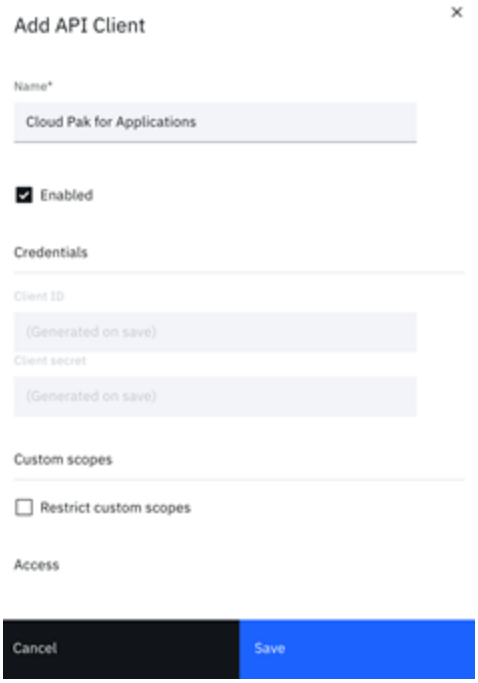
3. In the Configuration section, navigate to API Access > API clients where you can add an API Client.



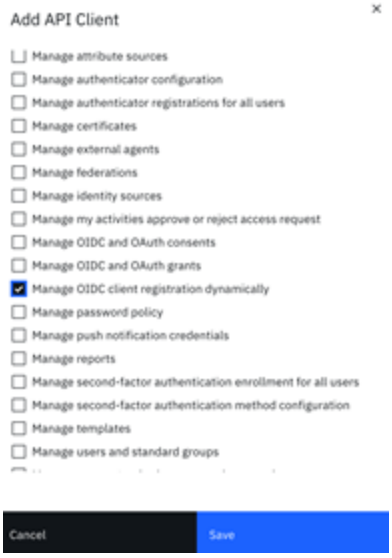
4. Click Add API Client.



5. Name and configure the new API Client.

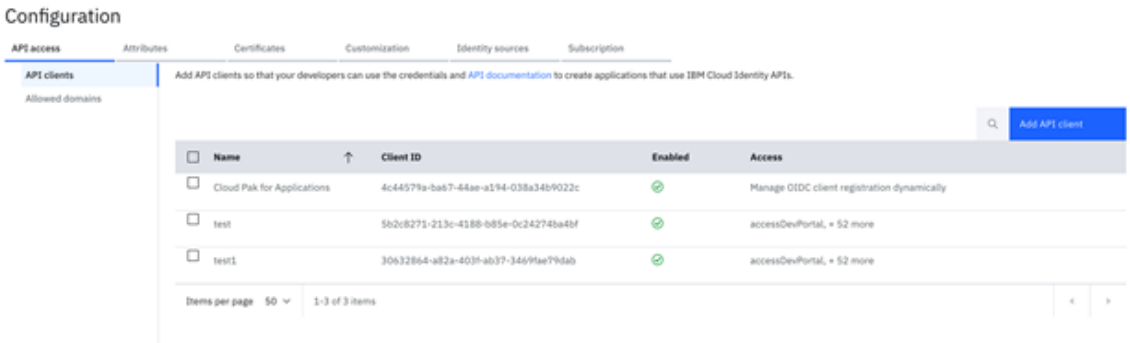


6. Select Manage OIDC client registration dynamically to ensure that applications are automatically registered with IBM Security Verify as the identity provider. This setting simplifies the experiences for application registration with product runtime operators.

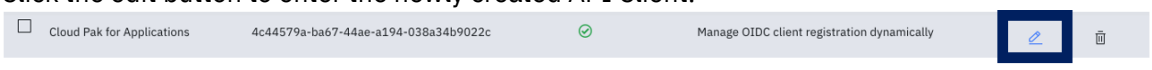


Other API permissions can be configured. For more information, see Managing API access.

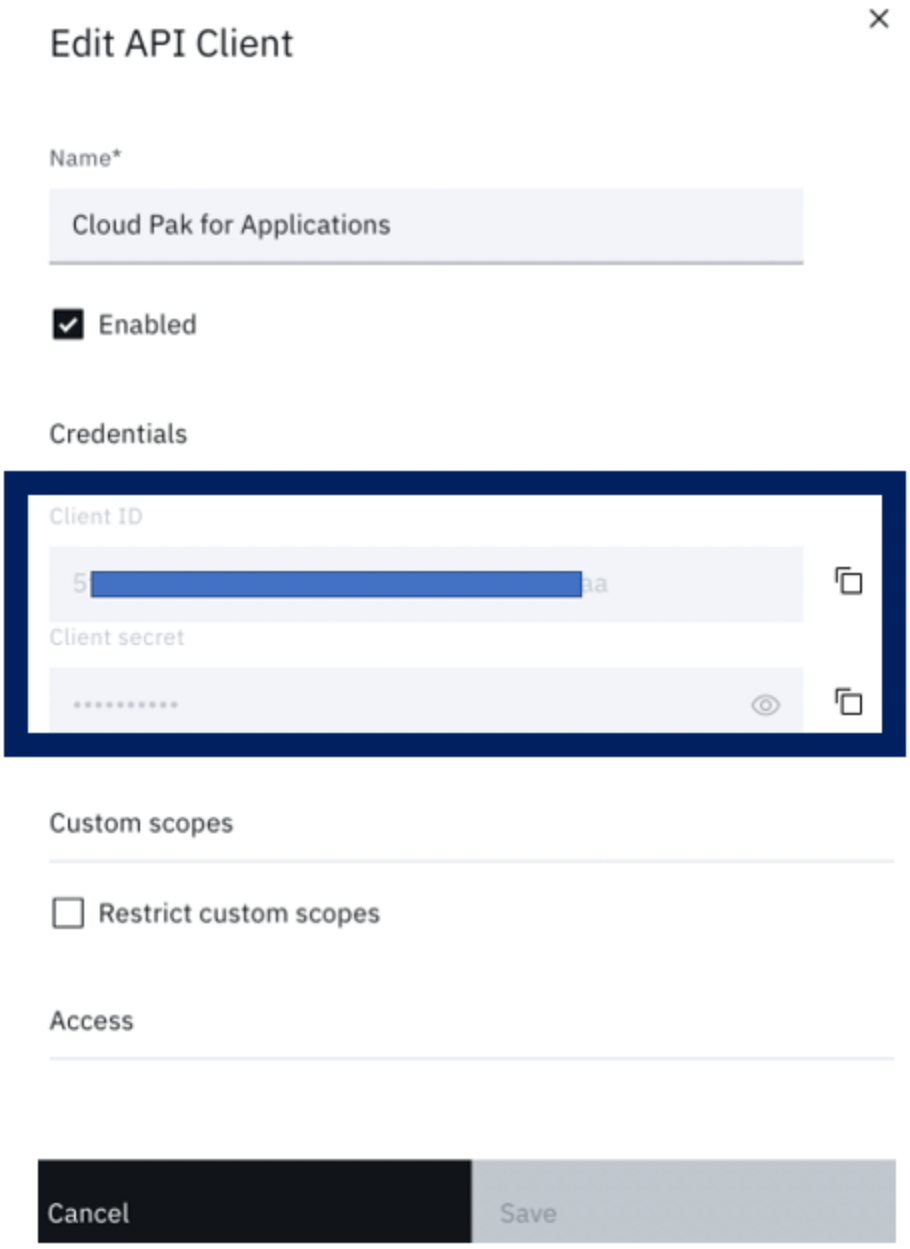
7. Save the new API Client. Verify that it is added to the API access table.



8. Click the edit button to enter the newly created API Client.



9. When selecting the recently created API client, you can see the client ID and secret that are needed to configure IBM Security Verify with WebSphere Hybrid Edition. The copy buttons next to each field make it easier to grab the data to paste into WebSphere Hybrid Edition Custom Resource Definitions (CRDs).



10. In addition to the Client ID and Secret that are associated with your API Client, you must add your *Discovery Endpoint URL* to your product custom resource definitions. To construct your *Discovery Endpoint URL*, append your *IBM Security Verify Tenant ID* to the beginning of the following generic *Discovery Endpoint URL*:

    `.ice.ibmcloud.com/oidc/endpoint/default/.well-known/openid-configuration.`

    You can find your *IBM Security Verify Tenant ID* in the address bar of your browser when you are logged in to IBM Security Verify. Your *IBM Security Verify Tenant ID* is the leading text string in the address bar, directly preceding **`.ice`**. For example, if the URL of your IBM Security Verify address is **`tenant-id.ice.ibmcloud.com/ui/admin`**, your *IBM Security Verify Tenant ID* is **`tenant-id`**, then your *Discovery Endpoint URL* is **`tenant-`**

id.ice.ibmcloud.com/oidc/endpoint/default/.well-known/openid-configuration.

In the following example IBM Security Verify URL, ***xxxxxx*** is the text that you append to the beginning of the generic tenant endpoint URL to construct the full tenant endpoint URL of ***xxxxxx*.ice.ibmcloud.com/oidc/endpoint/default/.well-known/openid-configuration**.

11. The data elements that you need when you configure the operator are:
    - Discovery Endpoint URL, which can be constructed from the tenant ID from your IBM Security Verify URL, along with the other items defined in step #10.
    - Client ID and secret, which can be found in step #9.
12. Provide the details in step #9 and #10 to the corresponding developers that might need to configure their application images with an OIDC IdP.

For developers:

1. Open your Red Hat OpenShift portal or command line.
2. Create an OpenLibertyApplication custom resource and enter your Discovery Endpoint URL. You will get your Discovery Endpoint URL from step #10 or from your administrator. For more information, see the OpenLiberty operator documentation.

```
    apiVersion: openliberty.io/v1beta1
    kind: OpenLibertyApplication
    metadata:
      name: test1-app
    spec:
      replicas: 1
      applicationImage: image-registry.openshift-image-
registry.svc:5000/test1/rp1
      expose: true
      service:
        port: 9443
      route:
        termination: passthrough
      sso:
        oidc:
          - discoveryEndpoint: <discovery-endpoint-url>
            autoRegisterSecret: my-autoreg-secret-olapp-sso
```

For example:

```
    apiVersion: openliberty.io/v1beta1
    kind: OpenLibertyApplication
    metadata:
      name: test1-app
    spec:
      replicas: 1
      applicationImage: image-registry.openshift-image-
registry.svc:5000/test1/rp1
      expose: true
      service:
        port: 9443
      route:
        termination: passthrough
```

```
      sso:
        oidc:
          - discoveryEndpoint: <tenant-
id>.ice.ibmcloud.com/oidc/endpoint/default/.well-known/openid-
configuration
            autoRegisterSecret: my-autoreg-secret-olapp-sso
```

3. Create a Kubernetes secret that holds the client ID and secret that are associated with IBM Security Verify that you captured in step #9 or that is provided by your administrator. For more information, see the OpenLiberty operator documentation.

```
    apiVersion: v1
    kind: Secret
    metadata:
      name: my-autoreg-secret-olapp-sso
      # Secret must be created in the same namespace as the
OpenLibertyApplication instance
      namespace: demo
    type: Opaque
    data:
      # base64 encode the data before entering it here.
      #IBM Security Verify requires a special clientId and clientSecret
for registration, and the registration URL.
       clientId: <client_ID>
       clientSecret: <secret>
```

For example:

```
    apiVersion: v1
    kind: Secret
    metadata:
      name: my-autoreg-secret-olapp-sso
      # Secret must be created in the same namespace as the
OpenLibertyApplication instance
      namespace: demo
    type: Opaque
    data:
      # base64 encode the data before entering it here.
      #IBM Security Verify requires a special clientId and clientSecret
for registration, and the registration URL.
       clientId: 4c94878a-ba69-44ae-a164-038j84b9022m
       clientSecret: y7fwHBvDBJ
```

What to do next

Administrators of the WebSphere Hybrid Edition environment can now create credentials for developers to start using IBM Security Verify as the IdP for developing applications locally or on cloud.

- To add your developers into the IBM Security Verify tenant, see the Managing users.
- To configure your application with IBM Security Verify, see Managing your applications.
- To configure the policies on your applications, see Managing access policies.

All tenants of IBM Security Verify (free or paid) are also entitled to IBM Application Gateway (IAG). This provides a proxy to help bridge the journey to cloud. IAG can help make the journey to cloud easier for legacy applications using non-OIDC authentication flows.

Open Liberty authentication

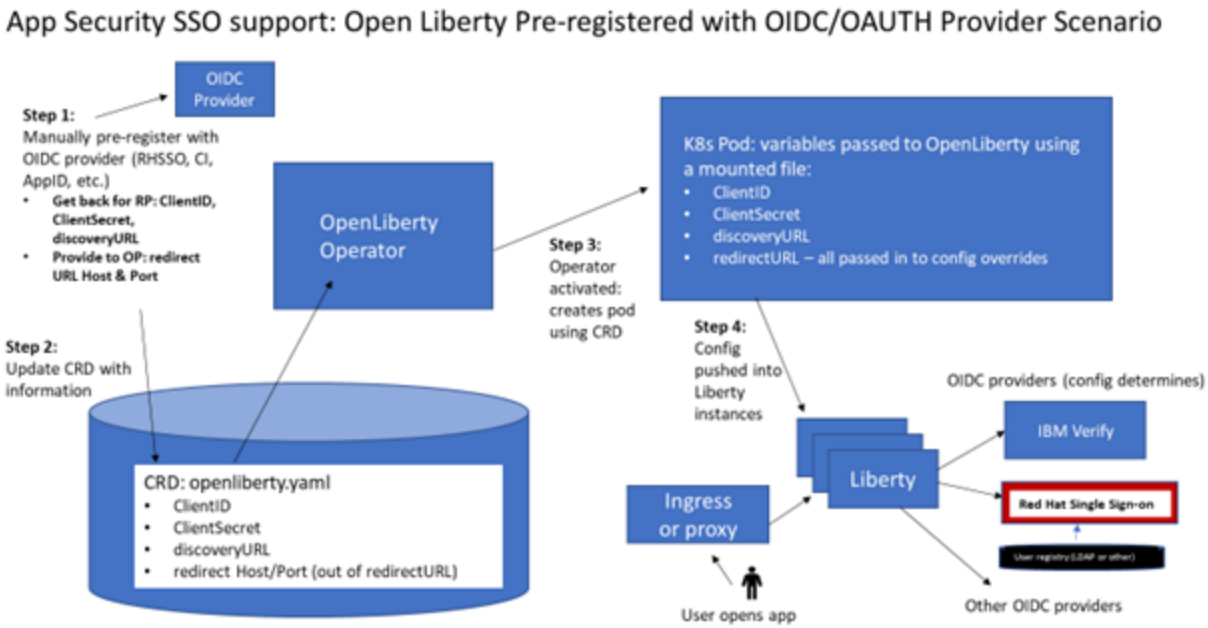
The Open Liberty operator in Red Hat OpenShift allows you to easily configure and manage the single sign-on with OpenID Connect or social medium for your applications.

- **Configuring identity providers**
  Open Liberty supports single sign-on by using the `socialLogin-1.0` feature with identity providers that is built on the standard protocols of OpenID Connect and OAuth. With the `socialLogin-1.0` feature, your application users can log in using their existing accounts for social media providers such as Google, Facebook, LinkedIn, Twitter, GitHub, or any OpenID Connect (OIDC) or OAuth 2.0 server account.
- **Configuring OpenID Connect by dynamic registration**
  The Open Liberty operator enables you to provision Open Liberty as an OpenID Connect relying party. The operator provides options for requesting an initial access token, which the runtime component operator uses to register Open Liberty and a Liberty instance as relying parties.

Configuring identity providers

Open Liberty supports single sign-on by using the `socialLogin-1.0` feature with identity providers that is built on the standard protocols of OpenID Connect and OAuth. With the `socialLogin-1.0` feature, your application users can log in using their existing accounts for social media providers such as Google, Facebook, LinkedIn, Twitter, GitHub, or any OpenID Connect (OIDC) or OAuth 2.0 server account.

Open Liberty operator allows to easily configure and manage the single sign-on information for your applications.



The configuration that is needed at image build time includes:

- The environment variable SEC_SSO_PROVIDERS must be defined and must contain a space delimited list of the identity providers to use. If more than one is specified, the user can choose which one to authenticate with. Any of the following values are valid: `oidc oauth2 facebook twitter github google linkedin`. Specify `ARG SEC_SSO_PROVIDERS="(your choice goes here)"` in your Dockerfile.
- Providers usually require the use of HTTPS. Specify `ARG TLS=true` in your Dockerfile.

- Your Dockerfile must call the configure.sh file for these to take effect.

Configuration for image build time or container deploy time

Since HTTPS is usually required, the following settings can simplify setup:

- To automatically trust certificates from well known identity providers:

  ```
  ENV SEC_TLS_TRUSTDEFAULTCERTS=true
  ```

- To automatically trust certificates issued by the Kubernetes cluster:

  ```
  ENV SEC_IMPORT_K8S_CERTS=true
  ```

Each single sign-on provider needs some additional configuration to be functional: a client ID, a client secret, and possibly more. These variables can be supplied in several ways:

- At build time, the variables can be defined in a server.xml file.

  ```
  <variable name="foo" value="bar" />
  ```

- At build time, the variables can be defined as `ENV` variables in the Dockerfile, although this is less secure.

  ```
  ENV name=value
  ```

- The variables can be passed as environment variables to the Docker container when it is deployed.
- The variables can be supplied in a deployment YAML file or by the Liberty operator, which pass them to the container during deployment.

Client ID and Client Secret are obtained from the provider. The RedirectToRPHostAndPort (SEC_SSO_REDIRECTTORPHOSTANDPORT) parameter is the protocol, host, and port that the provider should send the browser back to after authentication. For example:

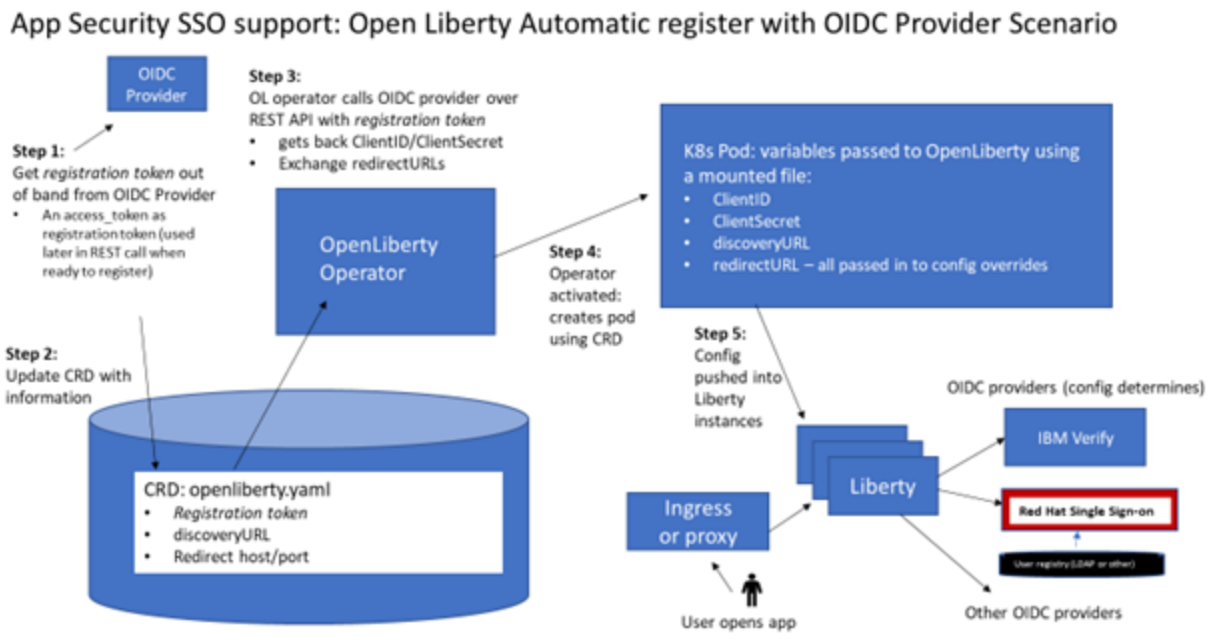
```
https://myApp-myNamespace-myClusterHostname.example.com
```

In some container environments, the pod cannot discern this and therefore it must be specified.

Other variables might be needed in some situations and are documented in detail in the Open Liberty documentation under each type of provider. The `oidc` and `oauth2` configurations are general purpose configurations for use with any provider that uses the OpenID Connect 1.0 or OAuth 2.0 specifications.

Configuring OpenID Connect by dynamic registration

The Open Liberty operator enables you to provision Open Liberty as an OpenID Connect relying party. The operator provides options for requesting an initial access token, which the runtime component operator uses to register Open Liberty and a Liberty instance as relying parties.



To configure dynamic registration, follow the identity provider configuration in the previous section, but instead of providing OpenID connect **`client_id`** and **`client_secret`**, you provide a registration token.

The operator can request a client ID and client secret from providers, rather than requiring them in advance. This can simplify deployment, as the provider's administrator can supply the information needed for registration once, instead of client IDs and secrets repetitively. An additional secret specified by the **`sso.oidc[].autoRegisterSecret`** function contains the information necessary to perform the registration. Automatic registration takes precedence over the values in the SSO secret. This is tested with Red Hat® Single Sign-on (RH-SSO).

```
apiVersion: v1
kind: Secret
metadata:
  name: my-autoreg-secret
  # Secret must be created in the same namespace as the OpenLibertyApplication
instance
  namespace: demo
type: Opaque
data:
  # base64 encode the data before entering it here.
  # RHSSO requires an initial access token for registration
  initialAccessToken: xxxxxyyyyy
  # others may require a dedicated clientId and clientSecret for registration
  clientId: bW9vb29vb28=
  clientSecret: dGhlbGF1Z2hpbmdjb3c=
  #
  # Optional: Grant types are the types of OAuth flows the resulting clients will
allow
  # Default is authorization_code,refresh_token.  Specify a comma separated list.
  # grantTypes: base64 data goes here
  #
  # Optional: Scopes limit the types of information about the user that the
provider will return.
  # Default is openid,profile.  Specify a comma-separated list.
  # scopes: base64 data goes here
  #
  # Optional: To skip TLS certificate checking with the provider, specify
insecureTLS as true.  Default is false.
  # insecureTLS: dHJ1ZQ==
```

Platform Application authentication

Your applications can be authenticated with Red Hat® OpenShift® native authentication service or IBM Cloud Pak foundational services Identity and Access Management service, and does not need provide its own authentication.

- **Authenticating with OAuth Proxy**
  The Red Hat OpenShift OAuth proxy is best used as a sidecar container in a Kubernetes pod, protecting a platform service that listens only on localhost.
- **Authenticating with Red Hat OpenShift Service Account Token**
  The Liberty `socialLogin-1.0` feature can be configured to use Red Hat OpenShift service accounts to authenticate and authorize protected resource requests.
- **Authenticating with OpenShift OAuth Server**
  The Social Login feature `socialLogin-1.0` can be configured to use the OAuth server and OAuth Proxy sidecar that are built-in to Red Hat OpenShift as authentication providers.
- **Setting up IBM Cloud Pak foundational services Identity and Access Management as OpenShift OAuth identity provider**
  The IBM Cloud Pak foundational services Identity and Access Management (IAM) feature provides a single-sign-on capability for Red Hat OpenShift authentication. To configure the IAM to handle Red Hat OpenShift authentication, first register an OpenID Connect client, and then create an OpenID Connect identify provider by using IAM for authentication.

Authenticating with OAuth Proxy

The Red Hat® OpenShift® OAuth proxy is best used as a sidecar container in a Kubernetes pod, protecting a platform service that listens only on localhost.

The OAuth proxy can be configured to allow user login with the same identity providers in OAuth server, or to authenticate clients with a service account token or OAuth token.

For more information, see https://github.com/openshift/oauth-proxy.

Authenticating with Red Hat OpenShift Service Account Token

The Liberty `socialLogin-1.0` feature can be configured to use Red Hat® OpenShift® service accounts to authenticate and authorize protected resource requests.

Using the `socialLogin-1.0` feature allows server administrators to secure, for example, monitoring and metrics endpoints that might produce sensitive information but require repeated access by an automated process or non-human entity. The new behavior allows service accounts to authenticate themselves by providing in the request a service account token that was created within the Red Hat OpenShift cluster.

A new `<okdServiceLogin>` configuration element is now provided to support this behavior. The `socialLogin-1.0` feature must be enabled to gain access to this new element.

The minimum configuration requires only that an **<okdServiceLogin>** element be specified in the server.xml file.

```
<server>

<!-- Enable features -->
<featureManager>
  <feature>appSecurity-3.0</feature>
  <feature>socialLogin-1.0</feature>
</featureManager>

<okdServiceLogin />

</server>
```

The minimum configuration assumes that the Liberty server is packaged and deployed within a Red Hat OpenShift cluster. By default, the **<okdServiceLogin>** element is used to authenticate all protected resource requests that the Liberty server receives.

Incoming requests to protected resources must include a service account token. The token must be specified as a bearer token in the Authorization header of the request. The Liberty server will use the service account token to query information about the associated service account from the Red Hat OpenShift cluster. The Red Hat OpenShift project that the service account is in will be used as the group for the service account when making authorization decisions. The Red Hat OpenShift project name is concatenated with the name of the service account to create the user name.

If the Liberty server is not deployed within a Red Hat OpenShift cluster, set the userValidationApi attribute to the value for the appropriate User API endpoint in the Red Hat OpenShift cluster.

```
<okdServiceLogin
userValidationApi="https://cluster.domain.example.com/apis/user.openshift.io/v1/users/~" />
```

Multiple **<okdServiceLogin>** elements can be configured as long as each element has a unique ID attribute specified. In those cases, configure the authentication filters to ensure the appropriate endpoints are protected by a unique **<okdServiceLogin>** instance.

For more information about Red Hat OpenShift service accounts, see the Red Hat OpenShift documentation for Understanding and creating service accounts.

Authenticating with OpenShift OAuth Server

The Social Login feature **socialLogin-1.0** can be configured to use the OAuth server and OAuth Proxy sidecar that are built-in to Red Hat® OpenShift® as authentication providers.

The Social Login feature has several pre-configured providers (such as Google, GitHub, or Facebook), but you can also configure additional providers (for example, Instagram, the Red Hat OpenShift OAuth server, and OAuth Proxy sidecar). The first is a standard OAuth Authorization Code flow, where a web browser accessing an app running in Liberty is redirected to the Red Hat OpenShift OAuth server to authenticate. The second accepts an inbound token from the Red Hat OpenShift OAuth Proxy sidecar or obtained from an Red Hat OpenShift API call. This approach requires less cluster-specific configuration.

You can run Liberty in a pod, but in the Authorization Code flow, Liberty can run outside the Red Hat OpenShift cluster. In either mode, an optional JWT can be created for propagation to downstream services.

Using Red Hat OpenShift as a provider differs slightly from other OAuth providers, because it requires a service account token to obtain information about the OAuth tokens. Once the client ID, secret, and token have been obtained from Red Hat OpenShift, Liberty can be configured as shown here:

To enable the feature, add it to the server.xml file.

See the following example server configuration for using Red Hat OpenShift OAuth server.

```
<server description="social">

  <!-- Enable features -->
  <featureManager>
    <feature>appSecurity-3.0</feature>
    <feature>socialLogin-1.0</feature>
  </featureManager>

<logging traceSpecification="com.ibm.ws.security.*=all=enabled" maxFiles="8"
maxFileSize="200"/>

<httpEndpoint  id="_home_markdown_jenkins_workspace_Transform_in_SSNMZP_doc_sec-
loginwithocp_defaultHttpEndpoint" host="*" httpPort="8941" httpsPort="8946" >
<tcpOptions soReuseAddr="true" /> </httpEndpoint>


  <!-- specify your clientId, clientSecret and userApiToken as liberty variables
or environment variables -->
  <oauth2Login id="_home_markdown_jenkins_workspace_Transform_in_SSNMZP_doc_sec-
loginwithocp_openshiftLogin"
    scope="user:full"
    clientId="${myclientId}"
    clientSecret="${myclientSecret}"
    authorizationEndpoint="https://oauth-
openshift.apps.papains.os.example.com/oauth/authorize"
    tokenEndpoint="https://oauth-
openshift.apps.papains.os.example.com/oauth/token"
    userNameAttribute="username"
    groupNameAttribute="groups"
    userApiToken="${serviceAccountToken}"
    userApiType="kube"

userApi="https://api.papains.os.example.com:6443/apis/authentication.k8s.io/v1/tok
enreviews">
  </oauth2Login>

  <keyStore id="_home_markdown_jenkins_workspace_Transform_in_SSNMZP_doc_sec-
loginwithocp_defaultKeyStore" password="keyspass" />

  <!-- more application config would go here -->

</server>
```

In the sidecar scenario, the configuration changes to accept an inbound token from the sidecar. See the following example server configuration for using the OAuth proxy sidecar:

```
<!-- specify your userApiToken as a liberty variable or environment variable -->
  <!-- note that no clientId or clientSecret are needed -->
  <oauth2Login id="_home_markdown_jenkins_workspace_Transform_in_SSNMZP_doc_sec-
loginwithocp_openshiftLogin"
    scope="user:full"
    userNameAttribute="username"
    groupNameAttribute="groups"
    userApiToken="${serviceAccountToken}"
    userApiType="kube"
```

```
    accessTokenHeaderName="X-Forwarded-Access-Token"
    accessTokenRequired="true"

userApi="https://kubernetes.default.svc/apis/authentication.k8s.io/v1/tokenreviews
">
  </oauth2Login>
```

To use HTTPS communication, either the server must have a key signed by a well-known certificate authority, which Liberty can trust automatically, or the server's public key must be added to the Liberty trust store. Red Hat OpenShift does not provide CA-signed keys by default, so the public key from Red Hat OpenShift OAuth server must be added. To add the public key, you can specify an environment variable in the server.env file. This setting identifies the file containing the public key in PEM format. Liberty reads the file and adds the key to its trust store.

```
# server.env

# OAuth sidecar scenario: causes the Kubernetes default certificate that is pre-
installed in pods to be added to Liberty trust store.
cert_defaultKeyStore=/var/run/secrets/kubernetes.io/serviceaccount/ca.crt

# OAuth server scenario: causes the public keys from /tmp/trustedcert.pem
(obtained separately) to be added to Liberty trust store.
cert_defaultKeyStore=/tmp/trustedcert.pem
```

Setting up IBM Cloud Pak foundational services Identity and Access Management as OpenShift OAuth identity provider

The IBM Cloud Pak foundational services Identity and Access Management (IAM) feature provides a single-sign-on capability for Red Hat® OpenShift® authentication. To configure the IAM to handle Red Hat OpenShift authentication, first register an OpenID Connect client, and then create an OpenID Connect identify provider by using IAM for authentication.

Before you begin

You must have a supported Red Hat OpenShift Container Platform version with IBM Cloud Pak foundational services installed, and have the following information available from Red Hat OpenShift and IBM Cloud Pak foundational services:

1. The admin user CS_DEFAULT_ADMIN_USERNAME and the admin password CS_DEFAULT_ADMIN_PASSWORD from IBM Cloud Pak foundational services installation.
   Tip: If you forget these values, you can retrieve them from the following `oc get secret` commands.

   ```
   oc -n ibmplatform-service get secret admin-credential -o
   jsonpath='{.data.defaultAdminUser}' | base64 --decode
   ```

```
oc -n ibmplatform-service get secret admin-credential -o
jsonpath='{.data.defaultAdminPassword}' | base64 --decode
```

The CS_DEFAULT_ADMIN_USERNAME is the default admin user for the OpenID Connect identity provider for IBM-IAM and must be unique to not conflict with a duplicate user name for another Identity provider.

2. The IBM Cloud Pak foundational services dashboard URL.
   Tip: If you forget the value, you can retrieve the URL with the following `oc get route` command.

```
oc -n kube-system get route icp-console -o=jsonpath={.spec.host}
```

3. The Red Hat OpenShift console URL.
   Tip: If you forget the value, you can retrieve the URL with the following `oc get route` command.

```
oc -n openshift-console get route console -o=jsonpath={.spec.host}
```

4. The issuer certificate authority root certificate icp-console.pem from the IBM Cloud Pak foundational services dashboard URL.
   Tip: If you forget the value, certificate with the following `oc get secret` command.

```
oc -n kube-system get secret icp-management-ingress-tls-secret -
o=jsonpath='{.data.tls\.crt}' | base64 --decode | tee -a icp-console.pem
```

Register the OpenID Connect Client

To register an OpenID Connect Client, first create a Custom Resource (CR) that specifies the client information and add it to the cluster.

1. To register the Client custom resource, create a file named ocpclient.yaml.

```
apiVersion: oidc.security.ibm.com/v1
kind: Client
metadata:
 namespace: default
 name: ocpclient
spec:
 secret: ocpclientsecret
 oidcLibertyClient:
   post_logout_redirect_uris:
   - https://console-openshift-console.apps.example.com:443
   trusted_uri_prefixes:
   - https://console-openshift-console.apps.example.com:443
   redirect_uris:
   - https://oauth-openshift.apps.example.com/oauth2callback/IBM-IAM
```

2. Modify the ocpclient.yaml contents to match your Red Hat OpenShift cluster environment.
   - Set the values of the post_logout_redirect_uris and trusted_uri_prefixes parameters to your Red Hat OpenShift console URL with port 443. For example: `https://console-openshift-console.apps.example.com:443`
   - Set the value of the redirect_uris parameter to `https://oauth-openshift.apps.example.com/oauth2callback/IBM-IAM` (where `example.com` matches your environment).

   Note: `IBM-IAM` is the name of OpenID Connect identity provider that is created in the Create the OpenID Connect Identity Provider for IAM authentication section.
3. To create the Client Custom Resource, run the following `oc create` command.

```
oc create -f ocpclient.yaml
```

You should see the `client.oidc.security.ibm.com/ocpclient created` message.

Note: This generates the **ocpclientsecret** secret, which stores the client ID and client secret.

4. To verify that the client registration is successful, run the following **oc get** command.

```
oc get Client ocpclient -n default
```

```
NAME         SECRET             READY   AGE
ocpclient    ocpclientsecret    True    7s
```

You should see the client status of READY equal to `True`.

5. To get the CLIENT_ID and CLIENT_SECRET from the **ocpclientsecret** secret to be used when creating the identity provider, run the following commands:

```
oc -n default get secret ocpclientsecret -o jsonpath='{.data.CLIENT_ID}' |
base64 --decode
```

```
oc -n default get secret ocpclientsecret -o jsonpath='{.data.CLIENT_SECRET}'
| base64 --decode
```

Create the OpenID Connect identity provider for IAM authentication

To create an identity provider, first create a Custom Resource (CR) that specifies the identity provider and add it to the cluster.

To configure the OpenID Connect identity provider from the Red Hat OpenShift console:

1. Select Administration > Cluster Settings.
2. In Cluster Settings, select the Global Configuration tab, and the OAuth Configuration Resource.
3. From OAuth Details, scroll down to the Identity Providers section.
4. Click Add and select OpenID Connect.
5. Specify the following values for your OpenID Connect identity provider for IAM:
   - Name: IBM-IAM
   - Client ID: **CLIENT_ID** that you retrieved previously.
   - Client Secret: **CLIENT_SECRET** that you retrieved previously.
   - Issuer URL: ***<IBM_Cloud_Pak_foundational_services_dashboard_URL>***`/oidc/endpoint/OP`. For example: **https://icp-console.apps.example.com/oidc/endpoint/OP**
   - Under More Options:
     - CA File: browse to import the issuer certificate authority icp-console.pem file that you retrieved in the Before you begin section.
     - Extra Scopes: **profile**
6. Click Add to create the OpenID Connect identity provider for IAM.

Log in to the Red Hat OpenShift console using IAM

The next time the Red Hat OpenShift console login page is presented, it allows the user to select **IBM-IAM** as the identity provider and the IBM Cloud Pak login page displays to prompt the user for username and password. The values for the CS_DEFAULT_ADMIN_USERNAME and CS_DEFAULT_ADMIN_PASSWORD parameters can be used to log in as the default administrative user.

Certificate management and TLS

You need configure applications to provide secured communications between a client and the server with transport layer security (TLS). To enable TLS, X509 certificate is required.

You can use certificate management functions provided by Red Hat OpenShift for your applications. This section highlights some common TLS and certificate management scenarios in WebSphere Hybrid Edition.

- **Configuring TLS for Liberty**
  The default Liberty server certificate is self-issued, so a client cannot verify the Liberty server certificate by default. Learn how to configure TLS for your applications.
- **Configuring TLS for platform applications**
  Platform applications can use Red Hat OpenShift service serving certificate to encrypt traffic.

Configuring TLS for Liberty

The default Liberty server certificate is self-issued, so a client cannot verify the Liberty server certificate by default. Learn how to configure TLS for your applications.

You can set the SEC_TLS_TRUSTDEFAULTCERTS environment variable to true to automatically trust certificates from well-known certificate authorities. You can set the SEC_IMPORT_K8S_CERTS environment variable to true to automatically trust certificates issued by the Kubernetes cluster. Alternately, you can include the necessary certificates manually when building an application image, or when mounting them using a volume when deploying your application.

Automatically trust known certificate authorities

To enable trust certificates from known certificate authorities, set the SEC_TLS_TRUSTDEFAULTCERTS environment variable. If set to true, then the default certificates from the JVM are used in addition to the configured truststore file to establish trust.

Providing custom certificates

It is possible to provide custom PEM certificates by mounting the files into the container. Files that are imported are tls.key, tls.crt and ca.crt.

The location can be specified by setting the TLS_DIR environment variable. The default location for certificates is /etc/x509/certs/.

The container automatically converts the PEM file and creates keystore and truststore files (key.p12 and trust.p12).

Container also can import certificates from Kubernetes. If the SEC_IMPORT_K8S_CERTS environment variable is set to true and the /var/run/secrets/kubernetes.io/serviceaccount folder is mounted into the container, the .crt files are imported into the the truststore file. Default value is **`false`**.

Providing a custom keystore

A custom keystore can be provided during the build phase of the application image by copying the keystore into the image's /output/resources/security/key.p12 location.

You must then override the keystore's password by including your copy of the keystore.xml file inside the /config/configDropins/defaults/ directory.

For instructions on configuring a secured service and a secured route with the necessary certificates, see Certificate manager integration.

Configuring TLS for platform applications

Platform applications can use Red Hat OpenShift service serving certificate to encrypt traffic.

Using the service serving certificate, you can significantly simplify the TLS and certificate management experience. The following steps describe a typical TLS scenario adopted in platform applications.

1. Request a service serving certificate for platform service from OpenShift `service-ca`.
2. Configure route or reverse proxy to terminate TLS connection.
3. Re-encrypt traffic with service serving certificate.

All internal traffics between platform services are re-encrypted with service serving certificates. All external traffics are over TLS with your certificate on route, and TLS terminates at the edge.

What to do next

For more information, see Securing service traffic using service serving certificate secrets.

Managing the environment

Liberty applications that run on Red Hat® OpenShift® can be managed using the resources at the following links.

- Application Logging on Red Hat OpenShift Container Platform (RHOCP) with Elasticsearch, Fluentd, and Kibana
- Application Monitoring on Red Hat OpenShift Container Platform (RHOCP) with Prometheus and Grafana

Getting help for WebSphere Hybrid Edition

If you experience problems with IBM WebSphere Hybrid Edition, you can receive assistance in several ways.

Finding answers to your questions

- Post your questions at StackOverflow for WebSphere Hybrid Edition.
- Request an invitation to the IBM Cloud Technology (Public Slack) channel to chat about your questions.

- Search for answers to your questions. Enter your search query in the IBM Documentation search field. Examine the search results on Documentation, Videos, IBM Developer, Technotes, and Redbooks tabs.

Reporting issues

- If you want to report a problem or are looking for support options, see WebSphere Hybrid Edition: Support Portal.
- To open a support ticket, follow the instructions in Opening a support case with IBM in the Support for WebSphere Hybrid Edition page.

Contacting sales

Contact sales with your queries.

1. Go to the WebSphere Hybrid Edition web page.
2. Click Let's talk.
3. Select to Call sales, Email sales, Book a meeting to schedule a consultation with an IBM expert, or Chat with sales.

- **Support for WebSphere Hybrid Edition**
  Learn how to get notifications, support, and information about support lifecycle policies for WebSphere Hybrid Edition. IBM Support is staffed by both IBM and Red Hat personnel, so you can get support for your entitled IBM and Red Hat products.

Support for WebSphere Hybrid Edition

Learn how to get notifications, support, and information about support lifecycle policies for WebSphere Hybrid Edition. IBM Support is staffed by both IBM and Red Hat® personnel, so you can get support for your entitled IBM and Red Hat products.

Signing up for notifications about WebSphere Hybrid Edition

You can be informed of critical IBM software support updates by using the My Notifications subscription service. For more information, see Stay up to date with My Notifications on the IBM website.

To subscribe to products of your choice, see My Notifications.

Opening a support case with IBM

Before you contact IBM Support, see the IBM Support Guide and the Getting Started Guide for IBM Support.

To open a support ticket, you must have an active entitlement for WebSphere Hybrid Edition. If you do not have an active entitlement, see Ordering an IBM WebSphere Hybrid Edition offering for a list of offerings.

Before you open a support case, if you are running in a cluster, collect MustGather data about your cluster.

To open a support case with IBM, follow these steps.

1. Go to the IBM Support site.

2. From the menu bar on the header, click Open a case.
3. Log in with your IBMid and password.
4. Enter a meaningful Case Title that summarizes your problem.
5. Select IBM as the Product Manufacturer.
6. Select WebSphere Hybrid Edition as the Product for which you need assistance from the support team.
7. Select the appropriate Severity of the problem. For more information about problem severity, see IBM Enterprise Support Severity Definitions.

   Note: The case severity is based on the business impact of the problem. If you set the case severity as 1, you must be available 24 hours a day to work with IBM Support on the issue.
8. Select the Account that has the entitlement for WebSphere Hybrid Edition.
9. Provide a detailed Case Description of your problem. A detailed description can help support understand your problem more accurately and thus provide quicker solutions or answers. The following information is crucial:
   - WebSphere Hybrid Edition product version
   - Installation platform (VMware, Microsoft Azure, Amazon Web Services (AWS), IBM Cloud®)
   - Red Hat OpenShift Container Platform version
   - Steps to reproduce the issue
10. Collect information about your cluster. For more information about how to gather the necessary information, see Collecting support information about the cluster.
11. Upload the `tar.gz` file with the results of the diagnostic scan that you ran in the previous step in one of the following ways:
    - Upload a file during the case creation
    - Send large files to Enhanced Customer Data Repository (ECuRep)
12. Select the language preferences.
13. Click Submit a case.

Note: Open a case for each problem that you need assistance with. Do not add new issues to an existing case for which you are already engaged with the support team. Clearly and completely define the issue to reduce confusion for the support team.

Understanding support lifecycle policies

WebSphere Hybrid Edition has a modified IBM Continuous Delivery (CD) Lifecycle Policy.

The numbering scheme for the version of WebSphere Hybrid Edition is based on the semantic version specification that is defined at: http://semver.org/.

The version numbers take the form `major.minor.patch`, where:

MAJOR
: Number changes occur when significant changes are introduced between releases, including incompatible changes.

MINOR
: Number changes indicate a new release.

PATCH
: Number changes for fix updates.

WebSphere Hybrid Edition uses a modified continuous delivery lifecycle policy. WebSphere Hybrid Edition is composed of several bundled products (programs) and components. The bundled products and components can be used separately from one another, except where otherwise stated by the license. Individual versions or releases of the bundled products continue to follow their own individual upstream support lifecycles.

For more information, see https://www.ibm.com/support/pages/node/6953601.

For more information about IBM software lifecycle policies, see https://www.ibm.com/support/pages/node/718165.

Viewing support lifecycle policies for bundled products and components

To view the support lifecycle policies for the bundled products and components in WebSphere Hybrid Edition, click the corresponding links in the following table.

Table 1. Viewing support lifecycle policies for bundled products and components

Product or component	Support lifecycle policy
IBM Cloud Transformation Advisor	https://www.ibm.com/support/pages/node/6955523
IBM Mono2Micro	https://www.ibm.com/support/pages/node/6955507
WebSphere Application Server Network Deployment	https://www.ibm.com/support/pages/lifecycle/search?q=5724h88
WebSphere Application Server Liberty Core	https://www.ibm.com/support/pages/lifecycle/search?q=5725L29
WebSphere Application Server	https://www.ibm.com/support/pages/lifecycle/search?q=5724j08