Installing Black Duck using OpenShift

Version 2022.4.0

This edition of the *Installing Black Duck using OpenShift* refers to version 2022.4.0 of Black Duck.

This document created or updated on Tuesday, April 12, 2022.

Please send your comments and suggestions to:

Synopsys
800 District Avenue, Suite 201
Burlington, MA 01803-5061 USA

Contents

Black Duck documentation

The documentation for Black Duck consists of online help and these documents:

Title	File	Description
Release Notes	release_notes.pdf	Contains information about the new and improved features, resolved issues, and known issues in the current and previous releases.
Installing Black Duck using Docker Swarm	install_swarm.pdf	Contains information about installing and upgrading Black Duck using Docker Swarm.
Getting Started	getting_started.pdf	Provides first-time users with information on using Black Duck.
Scanning Best Practices	scanning_best_practices.pdf	Provides best practices for scanning.
Getting Started with the SDK	getting_started_sdk.pdf	Contains overview information and a sample use case.
Report Database	report_db.pdf	Contains information on using the report database.
User Guide	user_guide.pdf	Contains information on using Black Duck's UI.

The installation methods for installing Black Duck software in a Kubernetes or OpenShift environment are Synopsysctl and Helm. Click the following links to view the documentation.

- Helm is a package manager for Kubernetes that you can use to install Black Duck.
- Synopsysctl is a cloud-native administration command-line tool for deploying Black Duck software in Kubernetes and Red Hat OpenShift.

Black Duck integration documentation can be found on Confluence.

Customer support

If you have any problems with the software or the documentation, please contact Synopsys Customer Support.

You can contact Synopsys Support in several ways:

- Online: https://www.synopsys.com/software-integrity/support.html
- Phone: See the Contact Us section at the bottom of our support page to find your local phone number.

To open a support case, please log in to the Synopsys Software Integrity Community site at https://community.synopsys.com/s/contactsupport.

Another convenient resource available at all times is the online customer portal.

Synopsys Software Integrity Community

The Synopsys Software Integrity Community is our primary online resource for customer support, solutions, and information. The Community allows users to quickly and easily open support cases and monitor progress, learn important product information, search a knowledgebase, and gain insights from other Software Integrity Group (SIG) customers. The many features included in the Community center around the following collaborative actions:

- Connect – Open support cases and monitor their progress, as well as, monitor issues that require Engineering or Product Management assistance
- Learn – Insights and best practices from other SIG product users to allow you to learn valuable lessons from a diverse group of industry leading companies. In addition, the Customer Hub puts all the latest product news and updates from Synopsys at your fingertips, helping you to better utilize our products and services to maximize the value of open source within your organization.
- Solve – Quickly and easily get the answers you're seeking with the access to rich content and product knowledge from SIG experts and our Knowledgebase.
- Share – Collaborate and connect with Software Integrity Group staff and other customers to crowdsource solutions and share your thoughts on product direction.

Access the Customer Success Community. If you do not have an account or have trouble accessing the system, click here to get started, or send an email to community.manager@synopsys.com.

Training

Synopsys Software Integrity, Customer Education (SIG Edu) is a one-stop resource for all your Black Duck education needs. It provides you with 24x7 access to online training courses and how-to videos.

New videos and courses are added monthly.

At Synopsys Software Integrity, Customer Education (SIG Edu), you can:

- Learn at your own pace.
- Review courses as often as you wish.
- Take assessments to test your skills.
- Print certificates of completion to showcase your accomplishments.

Learn more at https://community.synopsys.com/s/education.

Chapter 1: Installing using Synopsysctl

OpenShift™ is an orchestration tool from Red Hat used for managing cloud workloads through containers.

Synopsysctl is a command line tool that assists in the deployment and management of Synopsys software in Kubernetes and OpenShift clusters. After synopsysctl is installed, you can leverage it to easily deploy and manage Synopsys software.

Click here for documentation about installing and using synopsysctl.

If you are a customer using Kubernetes and are using an install method other than synopsysclt, please contact Synopsys Customer Support for migration assistance.

> **Note:** For scalability sizing guidelines, see the Container Scalability section of the Black Duck Release Notes.

> **Caution: Do not delete data from the Black Duck database (bds_hub) unless directed to do so by a Synopsys Technical Support representative. Be sure to follow appropriate backup procedures. Deletion of data will cause errors ranging from UI problems to complete failure of Black Duck to start. Synopsys Technical Support cannot recreate deleted data. Synopsys will provide support on a best-effort basis if no backups are available.**

Chapter 2: Hardware requirements

As of version 2022.4.0, the minimum recommended hardware requirements increased from 7 CPU cores & 28.5GB of RAM to 12 CPU cores and 37 GB of RAM. You can still manually change the hardware requirements and resource allocation using the provided helm charts or swarm configuration files. However, the current Black Duck default templates match the new recommendations. For the short to medium term we expect Black Duck to still be able to run successfully on the lesser requirements, but this may change over time. In the future, Black Duck will no longer be officially supporting nor testing with the older hardware specification. Customers who do not have available hardware to meet the new requirements will either need to add additional hardware or modify the orchestration files manually before updating to 2022.4.0.

Name	Scans/Hour	Black Duck Services	PostgreSQL	Total
10sph	10	CPU: 12 core Memory: 30 GB	CPU: 2 core Memory: 8 GB	CPU: 14 core Memory: 38 GB
120sph	120	CPU: 13 core Memory: 46 GB	CPU: 4 core Memory: 16 GB	CPU: 17 core Memory: 62 GB
250sph	250	CPU: 17 core Memory: 118 GB	CPU: 6 core Memory: 24 GB	CPU: 23 core Memory: 142 GB
500sph	500	CPU: 28 core Memory: 210 GB	CPU: 10 core Memory: 40 GB	CPU: 38 core Memory: 250 GB
1000sph	1000	CPU: 47 core Memory: 411 GB	CPU: 18 core Memory: 72 GB	CPU: 65 core Memory: 483 GB
1500sph	1500	CPU: 66 core Memory: 597 GB	CPU: 26 core Memory: 104 GB	CPU: 92 core Memory: 701 GB
2000sph	2000	CPU: 66 core Memory: 597 GB	CPU: 34 core Memory: 136 GB	CPU: 100 core Memory: 733 GB

This new guidance is based current Black Duck 2022.2.0 architecture. It is possible this guidance will be further refined for subsequent releases. If you have any questions or concerns, please reach out to Product Management.

Note: The amount of required disk space is dependent on the number of projects being managed, so individual requirements can vary. Consider that each project requires approximately 200 MB.

Black Duck Software recommends monitoring disk utilization on Black Duck servers to prevent disks from reaching capacity which could cause issues with Black Duck.

BDBA scaling is done by adjusting the number of binaryscanner replicas and by adding PostgreSQL resources based on the expected number of binary scans per hour that will be performed. For every 15 binary scans per hour, add the following:

- One binaryscanner replica
- One CPU for PostgreSQL
- 4GB memory to PostgreSQL

If your anticipated scan rate is not a multiple of 15, round up. For example, 24 binary scans per hour would require the following:

- Two binaryscanner replicas,
- Two additional CPUs for PostgreSQL, and
- 8GB additional memory for PostgreSQL.

This guidance is valid when binary scans are 20% or less of the total scan volume (by count of scans).

Note: Installing Black Duck Alert requires 1 GB of additional memory.

Chapter 3: PostgreSQL versions

Black Duck 2022.2.0 supports new PostgreSQL features and functionality to improve the performance and reliability of the Black Duck service. As of Black Duck 2022.2.0, PostgreSQL container 11 is the currently supported version of PostgreSQL for the internal PostgreSQL container.

Black Duck 2022.2.0 requires PostgresSQL 11 migration. This Black Duck 2022.2.0 update migrates the internal Black Duck PostgreSQL database container to version 11 of PostgreSQL. If you use the database container and deploy on OpenShift, you need to run a one-time migration job as documented in the Black Duck release notes and installation guide.

> Note: For PostgreSQL sizing guidelines, see the PostgreSQL Settings section of the Black Duck Release Notes.

If you choose to run your own external PostgreSQL instance, Synopsys recommends PostgreSQL 13.4 (or later 13.x) for new installs.

> **Caution: Do not run antivirus scans on the PostgreSQL data directory. Antivirus software opens lots of files, puts locks on files, etc. Those things interfere with PostgresSQL operations. Specific errors vary by product but usually involve the inability of PostgresSQL to access its data files. One example is that PostgresSQL fails with "too many open files in the system."**

Significant enhancements in PostgreSQL 11 include:

- Improvements to partitioning functionality, including:
    - Add support for partitioning by a hash key
    - Add support for PRIMARY KEY, FOREIGN KEY, indexes, and triggers on partitioned tables
    - Allow creation of a "default" partition for storing data that does not match any of the remaining partitions
    - UPDATE statements that change a partition key column now cause affected rows to be moved to the appropriate partitions
    - Improve SELECT performance through enhanced partition elimination strategies during query planning and execution
- Improvements to parallelism, including:

    - CREATE INDEX can now use parallel processing while building a B-tree index
    - Parallelization is now possible in CREATE TABLE ... AS, CREATE MATERIALIZED VIEW, and certain queries using UNION
    - Parallelized hash joins and parallelized sequential scans now perform better
- SQL stored procedures that support embedded transactions
- Optional Just-in-Time (JIT) compilation for some SQL code, speeding evaluation of expressions
- Window functions now support all framing options shown in the SQL:2011 standard, including RANGE distance PRECEDING/FOLLOWING, GROUPS mode, and frame exclusion options
- Covering indexes can now be created using the INCLUDE clause of CREATE INDEX
- Many other useful performance improvements, including the ability to avoid a table, rewrite for ALTER TABLE ... ADD COLUMN with a non-null column default

Note the following differences from the previous CentOS container:

- The base image is Debian Buster.
- The container does not run as root.
- PG runs as uid=1001 gid=0(root) groups=0(root) by default.
- There is no system Postgres user (but there is still a Postgres database user).

  > Note: A database user must be explicitly given when running psql from within the container; e.g., PGUSER=blackduck psql bds_hub or psql -U blackduck bds_hub.

- The PG data directory is at /bitnami/postgresql/data.
- pg_hba.conf is in /bitnami/postgresql/conf.
- postgresql.conf is managed by bitnami.

  > Note: ALL configuration changes MUST be made with ALTER SYSTEM rather than by editing the config file.

- The PostgreSQL volume is mounted on /bitnami/postgresql.
- pg_hba.conf is no longer overwritten every time the container starts.

General Migration Process

The guidance here applies to upgrading from any PG 9.6 based Hub (releases prior to 2022.2.0) to 2022.2.0 or later.

1. The migration is performed by the new blackduck-postgres-upgrader container.
2. The folder layout of the PostgreSQL data volume is rearranged to make future PostgreSQL version

upgrades simpler.

3. The UID of the owner of the data volume is changed. The new default UID is 1001, but see the deployment-specific instructions.
4. pg_upgrade is run to migrate the PG 9.6 database to PG 11.
5. Plain ANALYZE is run on the PG 11 database to initialize query planner statistics.
6. blackduck-postgres-upgrader exits.

Chapter 4: Migrating on OpenShift with Helm

This migration replaces the use of a CentOS PostgreSQL container with a Synopsys-provided container. Also, the synopsys-init container is replaced with the blackduck-postgres-waiter container.

Important: Customers with non-core PostgreSQL extensions are STRONGLY encouraged to uninstall them before migrating and reinstall them after the migration completes successfully; otherwise, the migration is likely to fail.

Important: Customers with replication set up will need to follow the instructions in the pg_upgrade documentation BEFORE they migrate. If the preparations described there are not made, the migration will likely succeed, but the replication setup will break.

The migration is performed by a one-time job:

1. Stop Black Duck; e.g.,

   ```
   kubectl scale --replicas=0 -n <your_namespace> deployments --selector
   app=blackduck
   ```

2. Run the upgrade job; e.g.,

   ```
   helm upgrade <your_deployment_name> . -n <your_namespace> <your_normal_
   helm_options> --set status=Stopped --set runPostgresMigration=true
   ```

3. Restart Black Duck as normal with helm upgrade.

On plain Kubernetes, the container of the upgrade job will run as root unless overridden. However, the only requirement is that the job runs as the same UID as the owner of the PostgreSQL data volume (which is UID=26 by default).

On OpenShift, the upgrade job assumes that it will run with the same UID as the owner of the PostgreSQL data volume.