portworx

PDS Security White Paper

March 2023

Table of contents

Introduction

Security is paramount in building customer trust and confidence. Our products and services are built with security in mind. We follow industry leading secure development practices—including security by design, security threat modeling, penetration testing, and container scanning—using industry leading, commercially available tools to ensure our products provide the highest level of security to protect our customers' data.

Our security operations team continuously monitors for emerging threats and vulnerabilities and addresses them proactively. Further, our products go through internal and external audits—including various security and compliance certification assessments—to help our customers achieve their compliance needs. Refer to our Security and Trust Center for up-to-date information on our security certifications.

The purpose of this white paper is to provide an overview of the PDS architecture, the shared responsibility model, and additional security considerations for our customers. In addition to reading this security white paper, we encourage you to read our product documentation to understand more about product features, how-to guides, and best practices.

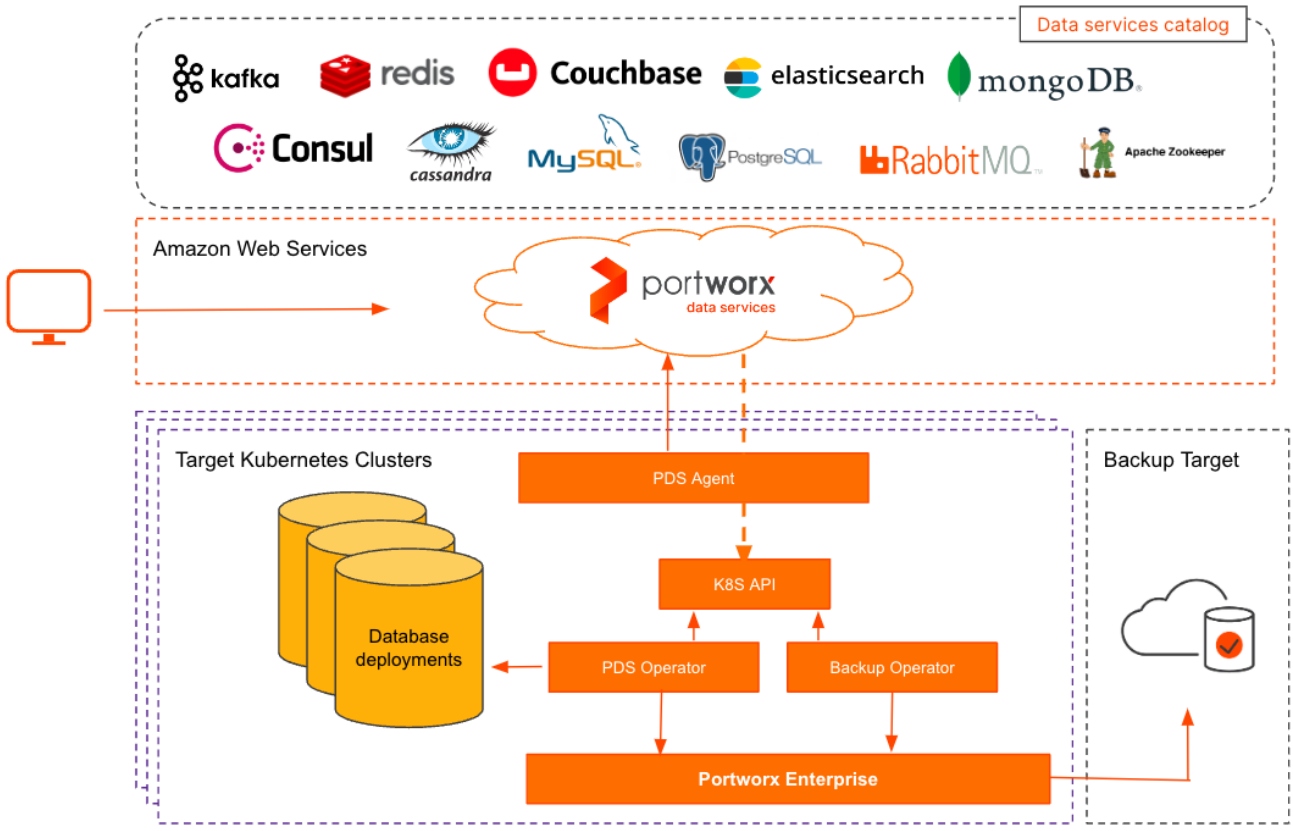
What Is the Portworx Data Services Platform?

Portworx Data Services (PDS) is a platform for running production-ready data services on Kubernetes clusters. These data services include relational databases, NoSQL databases, graph databases, key-value stores, document stores, message queues, search indexes, and event streaming services—such as Kafka, Redis, Cassandra, MySQL, RabbitMQ, etc. We are constantly working on expanding and supporting new data services.

PDS is offered as a platform as a service, providing a managed experience for our customers. This experience extends through the full lifecycle of data services, from initial deployment and through Day 2 operations and decommissioning. Portworx Data Services is built using cloud-native architecture consisting of a number of microservices and extensions to the Kubernetes API. Some of these components are delivered as software as a service, while others run in the customers' Kubernetes clusters.

PDS Architecture Overview

This Portworx architecture section provides an overview of the PDS components, the way they connect with your Kubernetes clusters, and the shared responsibilities between customers and Portworx.



PDS has been developed based on cloud-native architecture, and it is composed of the following components:

- **PDS Control Plane:** This is an interactive hosted SaaS service for the users to manage the PDS services on the target cluster. It provides the UI, API, and a catalog of data services. Customers interact with this control plane to deploy their data services on their Kubernetes clusters.
- **Data Services:** This includes relational databases, NoSQL databases, graph databases, key-value stores, document stores, message queues, search indexes,

and event streaming services such as Kafka, Redis, Cassandra, MySQL, RabbitMQ, etc. Portworx constantly adds new services to the catalog for customers to use.

- **Target Cluster:** This is the customer's Kubernetes cluster (on premises or cloud) where the data services are deployed.
- **PDS Operators:** A PDS Operator includes the following components that are deployed when connecting to the target cluster to facilitate management through the PDS control plane:
    - The **Deployment Operator** allows you to deploy the data service to the cluster and controls custom resources, services, config maps, secrets, etc.
    - The **Target Operator** is responsible for management of PDS Kubernetes and related third-party components, including their installation and updates. It is also intended to communicate the status of the components and other PDS-specific information to the control plane.
    - The **PX Operator** is responsible for the installation of Portworx, which serves as a backbone for management of persistent storage within the cluster.
    - The **Backup Operator** allows you to take backups, both logical and physical, of your data services and offload them to your backup target.
    - **Teleport** is a third-party library used to enable a secure connection between the control plane and target cluster.
    - **Prometheus**—currently deployed by Helm Chart but soon managed by the target operator in agent mode—helps scrape metrics and limits the amount of metrics exported by data services.
- **PDS Agent** (also known as Platform agent): This is responsible for registering Kubernetes clusters to the control plane and installing Teleport and PX Operator.
- **Backup Target:** This is a location on cloud—such as AWS S3, Azure Blob Storage, or Google Cloud—where the backup must be stored.

Shared Responsibility Model

Portworx Data Services (PDS) uses a shared responsibility model for security. Portworx has implemented reasonable security measures and controls necessary to protect the data and integrity of its services. Customers are responsible for securing other areas within their area of responsibility. The table below outlines the responsibility between Portworx and customers.

Areas of Responsibility	Portworx	Customers
Provisioning		
Deployment of Portworx Enterprise Storage (Prerequisite)		Y
Deployment of Kubernetes clusters/target cluster with adequate storage		Y
Persistent volumes in Kubernetes cluster for the PDS Data services	Y	
PDS Data Services (MySQL, Cassandra, Postgres, etc.)	Y	
Configuration, Patching, Vulnerability Management, and Updates		
Security of Kubernetes Clusters		Y
Providing the latest version of Data Services	Y	
Updating Data Services to latest versions		Y
Providing updates to target cluster PDS components	Y	
Updating PDS components on the target cluster		Y
Security of PDS control plane	Y	
Access Management		
Identity and Access to Kubernetes clusters		Y
User Access Management for Data Services		Y
User Access Management for PDS control plane		Y

Data Management & Security		
Customer data security (masking/encryption)		Y
Data Services hardening		Y
Data Services transaction logging/monitoring		Y
Availability		
Kubernetes cluster and PDS target cluster components		Y
PDS control plane	Y	

PDS Control Plane Security

The PDS control plane is built on top of Amazon Web Services(AWS). Portworx is responsible for the security, integrity, and availability of the PDS control plane and the UI.

All communication between a target cluster and the control plane is encrypted by default using TLS v1.2 or above.

Currently, PDS collects backup target credentials—such as Amazon S3, Azure, or GCP—and encrypts them at rest in the database. AWS Secrets Manager will be rolled out in the upcoming PDS release to store all the user credentials for enhanced security.

Customer Responsibility

The PDS control plane allows customers to self-manage their user accounts; add/remove additional users, manage roles, user API keys, and key rotation policies. The customer is responsible for ensuring that the access is restricted to appropriate individuals and for ongoing user access management.

Data Service Deployment and Security

Portworx deploys the following components onto the customer's Kubernetes cluster. Portworx is responsible for ensuring the security and integrity of these components when they are deployed.

- Data service images such as Kafka, Cassandra, MySQL, Redis, etc.
- Additional components—such as Portworx Operators and PDS agent—to manage data services

PDS Agent communicates with the PDS control plane using secure HTTP and gRPC and it creates a secure reverse tunnel for the PDS control plane to communicate with the target cluster's Kubernetes API server. This gives PDS Agent the capability to deploy services and schedule backups.

Access to the target cluster's Kubernetes API server is controlled by the local Kubernetes service account for the PDS Agent. It can be used to terminate the connection between the PDS control plane and target cluster, if needed.

Portworx provides updates to data service images periodically. This includes security vulnerabilities, bug fixes, or newer versions. Customers are responsible for updating the data services to the latest versions. Portworx also provides security updates to operators and agents that require customers to apply manually.

Customer Responsibility

When PDS deploys a data service on a customer's Kubernetes cluster, it creates a default user account called "pds." The customer is responsible for changing the default credentials for the "pds" account.

Additionally, PDS may create "pds-observability" or "pds-admin" accounts to collect DB metrics for observability. These accounts are required for metrics and dashboard functions to work, and they should not be modified or deleted.

Customers are responsible for adding additional accounts and user access management from this point.

The customer is responsible for ensuring that the Kubernetes service account that controls the target cluster's Kubernetes API is enabled at all times for the PDS Agent to communicate with the PDS control plane.

To enable application connection to the databases from outside the cluster if required, ExternalDNS interacts with the cloud provider to set up the necessary DNS records and assigns an IP address that may resolve publicly to the relevant services within the cluster, but cannot be routed to from the Internet.

Network Communications

This section describes network communication between the target cluster and PDS control plane.

- All network connections are egress from the target cluster. Therefore, no open ingress ports are required.
- All network connections are encrypted using TLS 1.2 or above and authenticated.
- Requests from the control plane to the target cluster don't establish their own connections but are tunneled through an existing secure reverse proxy connection.
- Each PDS domain has a set of static IP addresses that can be used for firewall rules.

Target	Domain	IP Addresses	Ports	Protocols	Authentication
PDS API server	prod.pds.portworx.com	100.20.37.112 44.227.81.37 50.112.214.0	443	HTTPS	Bearer token
Cortex	prod-observability.pds.portworx.com	34.210.13.239 44.225.179.27 52.40.228.95	443	HTTPS	Bearer token
Reverse Tunnel Server	prod-teleport.pds.portworx.com	44.233.212.136 52.88.139.36 54.191.49.17	443, 3024	HTTPS, gRPC, SSH	Join token, node certificate

Telemetry and Usage Data

PDS collects and processes information—such as configuration, performance, user activity, and diagnostics log data—from PDS components and Kubernetes APIs, operators, and agents. This information is processed to facilitate the delivery of cloud services, including providing product support, troubleshooting technical issues, tracking entitlements, improving user experience, and monitoring and ensuring the performance, integrity, and stability of the components and services. Customers should not name the instances, namespaces, or cluster names using confidential or personal data. To the extent that any of this data is considered personal data under applicable data protection laws, the data will be treated in accordance with our privacy policy and notices.

Customer Security Responsibilities

Target Cluster

The data services are deployed on the target Kubernetes cluster. The customer is responsible for managing and securing target clusters and keeping them up to date. Further, customers should ensure access to PDS components—such as the PDS operator, Backup operator, and PDS agent running on the target cluster is restricted to appropriate individuals.

Backup Target

Customers provide the object stores for backup targets and are responsible for keeping them secure. Customers are responsible for rotating backup target credentials on a periodic basis and updating the new credentials in the PDS control plane.

Portworx Security Overview

Portworx has established and implemented an Information Security Management System with a dedicated team responsible for information and product security. All Portworx

employees go through a background check as part of their hiring, and they are required to complete security and privacy training. Additionally, all employees are required to complete security awareness training at least annually. Our information security team conducts periodic security awareness campaigns and phishing tests to keep up with growing security threats and to improve awareness.

Access to Portworx's systems is provided to authorized individuals strictly on a need-to-know basis with least privilege principles. Access to critical systems is periodically reviewed for appropriateness and recertified by the system owners. Segregation of production and non-production environments is maintained to avoid accidental changes or modifications.

Information assets are continually monitored for threats, unexpected behaviors, anomalies, and vulnerabilities. The Incident Response Program promptly identifies, declares, and responds to security incidents.

Secure Development and DevSecOps

Portworx follows security by design principles for secure software development. All products and services go through a threat modeling exercise during the design phase to identify potential threats and bake in appropriate security controls. Engineers are required to go through secure coding training. Industry leading, commercially available tools have been implemented to scan the code and container images for security vulnerabilities and have been addressed appropriately during various stages of SDLC.

Penetration testing is performed on products and services periodically, and any issues identified are addressed in a timely manner. Appropriate security controls have been implemented for securely deploying and operating the service—such as web application firewalls, IDS/IPS, API security, AWS hardening, etc. Monitoring tools and processes have been established for monitoring and responding to security threats.

Additional Resources

PDS Product documentation (https://pds.docs.portworx.com/)

PDS Product Support (https://docs.portworx.com/support/)

Release Notes (https://pds.docs.portworx.com/release-notes/)

Security and Trust Center (https://portworx.com/security-and-trust-center/)

Privacy Policy (https://www.purestorage.com/privacy.html)