

What is Substrate

Downloaded from Epic Games Confluence

Date: 2025-07-12 04:05:23

Original URL: <https://confluence-epicgames.atlassian.net/wiki/spaces/CDE/pages/81068480>

Document Level Classification

100

- [Introduction](#)
- [Features](#)
- [Architecture](#)
- [Requesting a Substrate Environment](#)

Introduction

Substrate is Epic Games' Internal Cloud Developer Platform designed to provide teams with their own infrastructure for running workloads. Substrate is hosted on Amazon Web Services (AWS) and managed by the Cloud Engineering team.

Essentially Substrate is a pre-provisioned multi-account environment consisting of 2 AWS accounts (Live and Dev) with all of the infrastructure needed for hosting workloads, provisioned within them. Development

workloads are powered by AWS EKS Managed Kubernetes clusters. These pre-provisioned environments provide developers and/or service owners with all of the necessary infrastructure to get started with developing services quickly without having to worry about having to manually provision those resources themselves or have expert knowledge of AWS.

The substrate value proposition includes the integration of various products to offer a combination of defaults for accelerated service roll out compared to service owners individually selecting and building the entirety of their infrastructure. Service owners have great autonomy to select all or parts of our defaults. Default workflows are created with evolving preselected technology stacks, and a combination of internal and 3rd party products. Service owners have the freedom to customize default workflow details for their use cases and to influence their desired outcomes.

Most service owners are using most offered Infrastructure Platform features, and at a minimum running services on Substrate comes with underlying integration into Epic's information security, and policy based data and access segregation.

The full list of features and their description are provided below as well as the architectural diagram for a visual representation of what comes with a single substrate environment.

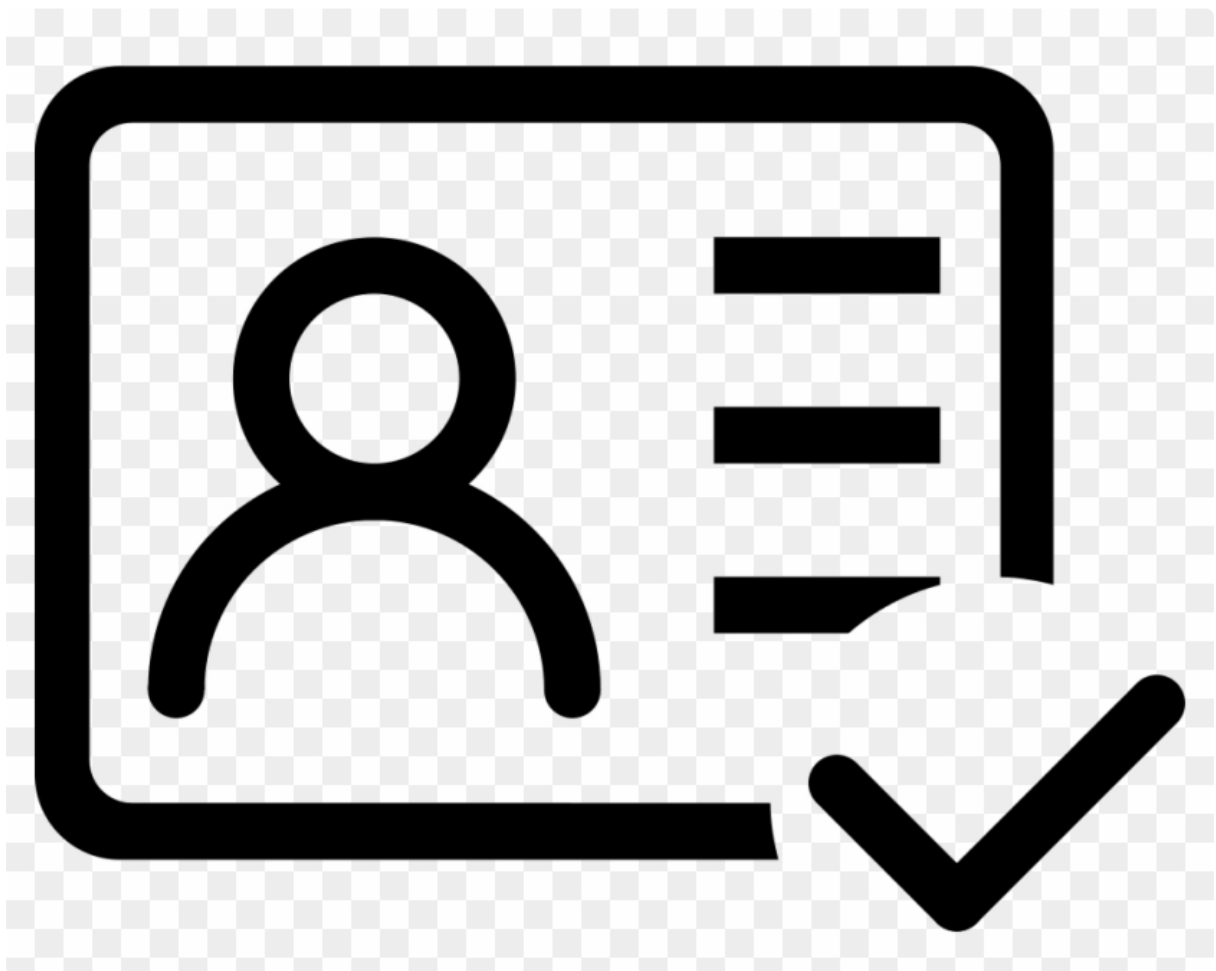
To get started with Substrate, review the [Getting started guide](#).

Features



AWS Account Creation

Managed account setup including billing and connection to the Epic AWS organization. Upon request of a new environment, development and production accounts are created to make separation of concerns easier.



Easy Account Authentication

A tool called *aop* is provided for easy AWS token retrieval and Kubernetes context creation. Access to EC2 and RDS happen through *Teleport*.

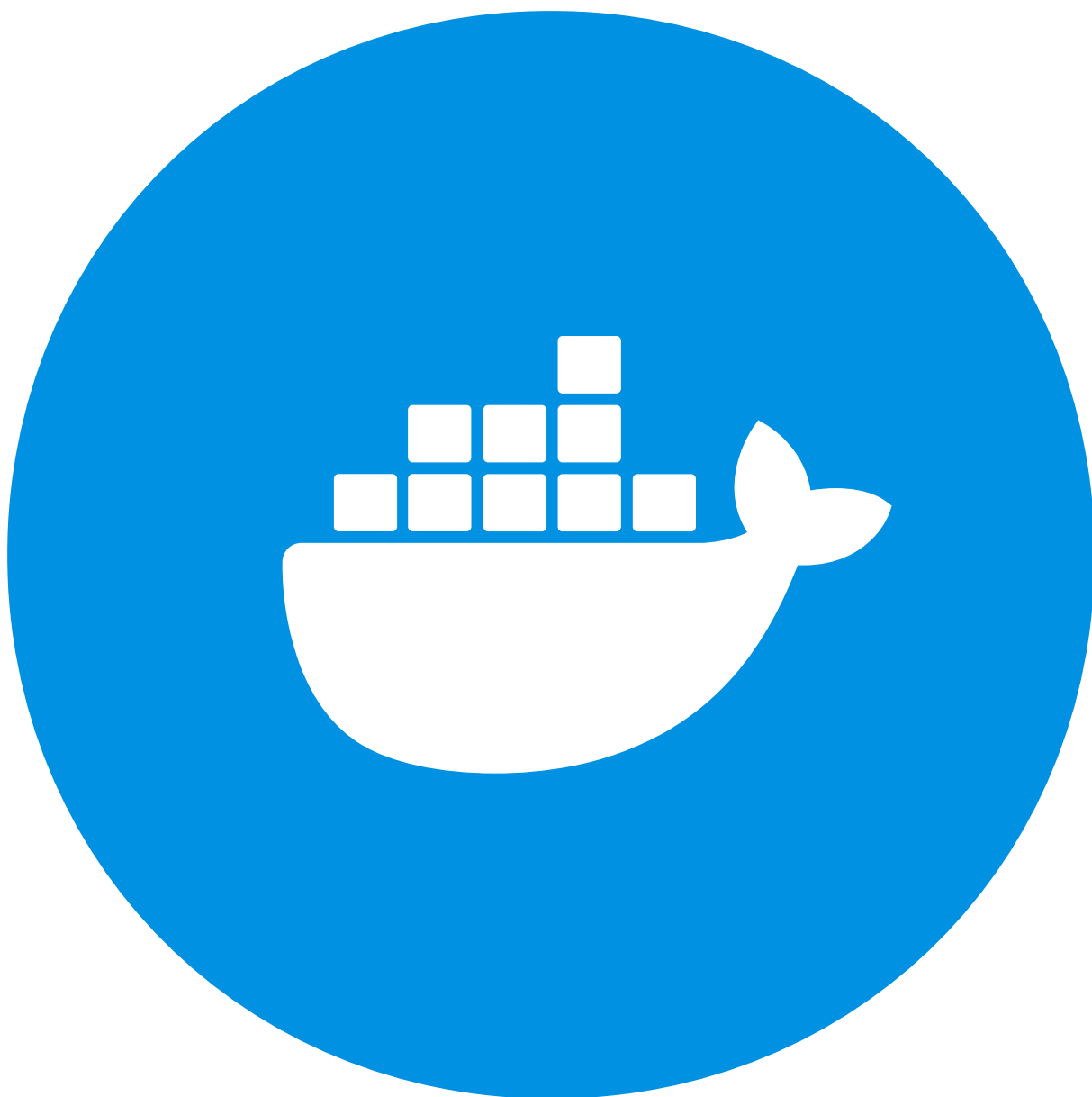


Cross-Account Connectivity

A [Service Network](#) for connecting to services in Oldprod, other Substrate accounts, as well as the Office and VPN accounts; is provided, upon request without worrying about IP conflicts and routing problems.



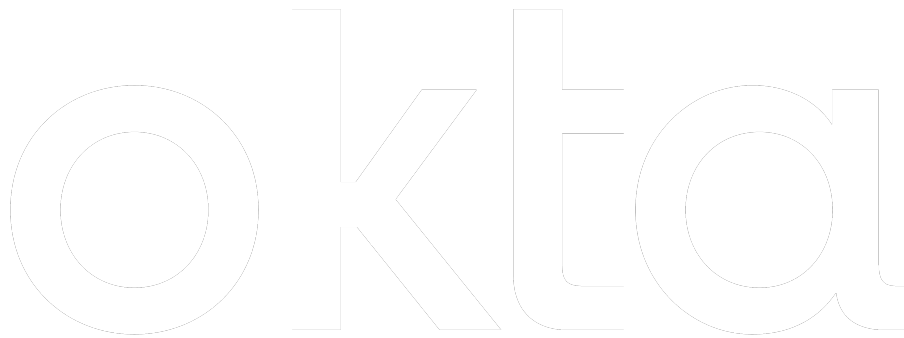






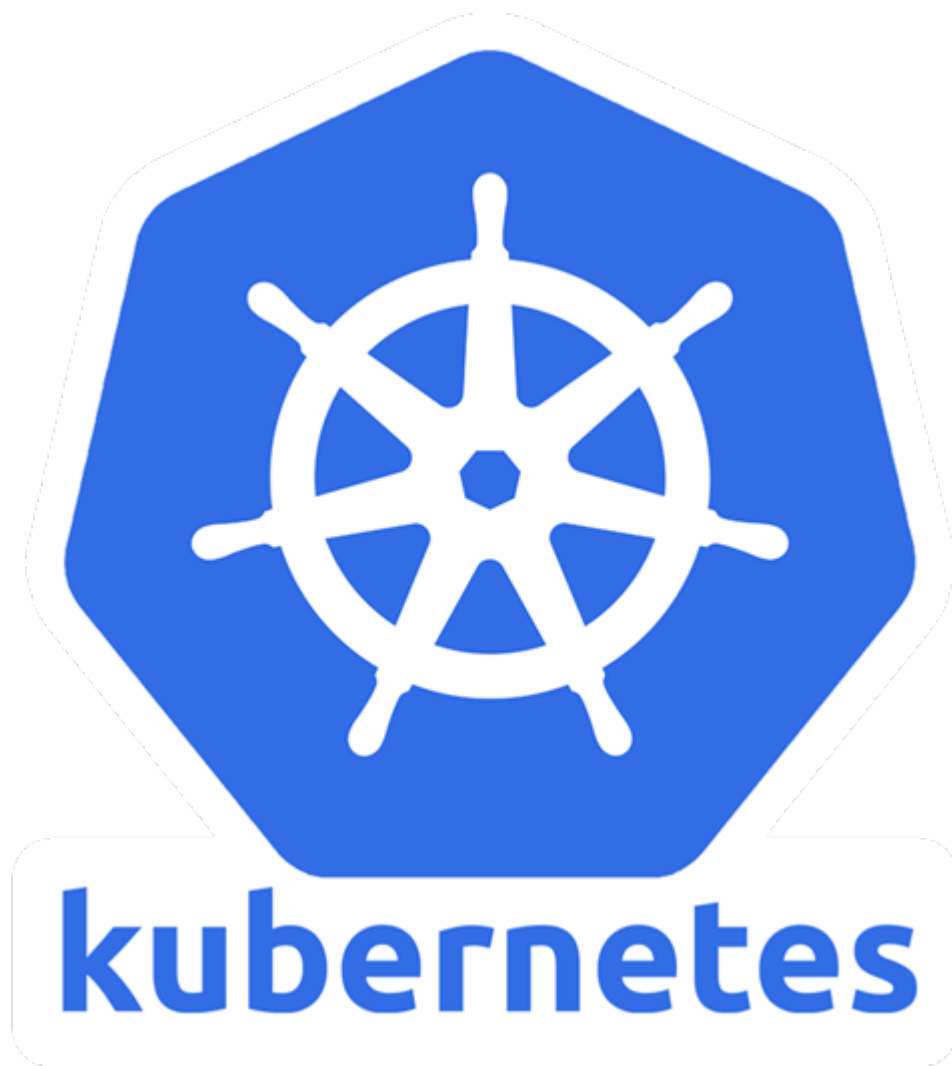
CI/CD Integration

Managed *Codefresh* (CI/CD) setup as well as its integration to your AWS account and EKS cluster. *Codefresh* is also configured so that it can access GitHub Enterprise, JFrog Artifactory, Vault, and more. Ongoing support is provided for other CI/CD tooling including *TeamCity* and *GitHub Actions*.



Simplified Access Management

Access to all related Substrate tools is granted systematically based on *Okta group memberships* and/or *SailPoint Roles* - tools that are managed by IT, reducing onboarding friction.



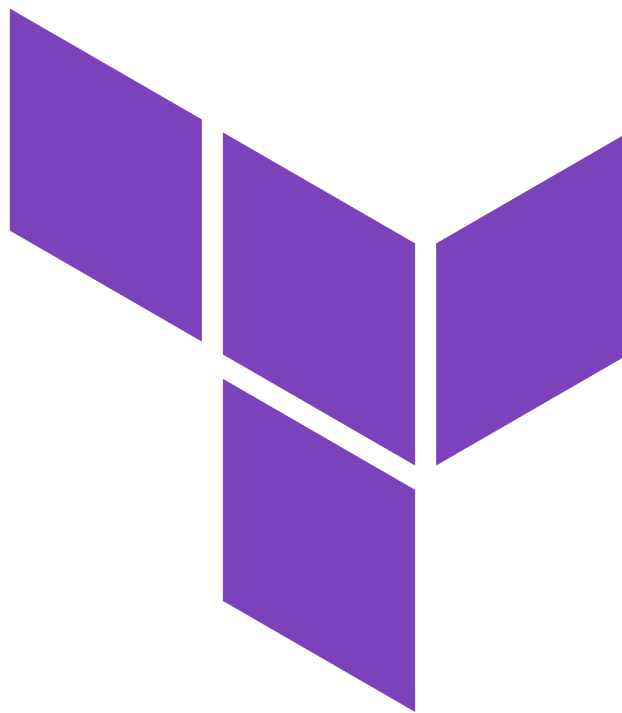
Managed Kubernetes

Amazon Elastic Kubernetes Service (Amazon EKS) clusters are deployed for your team and your initial namespaces and role-based access control (RBAC) are configured. It also includes white-glove upgrades to ensure that your cluster is not more than one version behind the latest Amazon EKS version. This includes checking for and notifying you of deployed Kubernetes resources that might break your CI/CD pipeline after an upgrade.



Build Artifact Repository

Access to *JFrog Artifactory* is provided to store project build artifacts from CI/CD tools such as *CodeFresh*, *TeamCity*, and *GitHub Actions*.



Infrastructure as Code (IaC)

Terraform Enterprise (TFE) access is provisioned by default, allowing you to deploy infrastructure resources to your account with standard IaC workflows.



Deployment Tooling

A *Helm* chart (epic-app) is provided, which makes it easy to deploy your applications to Kubernetes using our curated best practices. This chart is kept up to date and add features over time.

Secrets Management

A *Vault* instance is provided, as well as *Self Service Secrets Management (SSSM)*. A secret path is provisioned to your team in order to make your secrets available to *Codefresh*, your *EKS* clusters, and more. Patterns for configuring a *Vault* sidecar that fetches your secrets when your Pods start are provided.



Networking

When your AWS account is set up, various network resources are provided such as an *Internet Gateway (IGW)*, *Virtual Private Cloud (VPC)*, multiple subnets, routes, and *AWS Endpoints* for accessing *OldProd*. Routes and connectivity are set up for *GitHub Enterprise*, *Artifactory*, our internal *Helm* Chart repository and more.



DNS

Each substrate account gets a unique DNS zone under `*.on.epicgames.com` delegated to ensure that you have space to name your services without worrying about conflicting DNS entries.



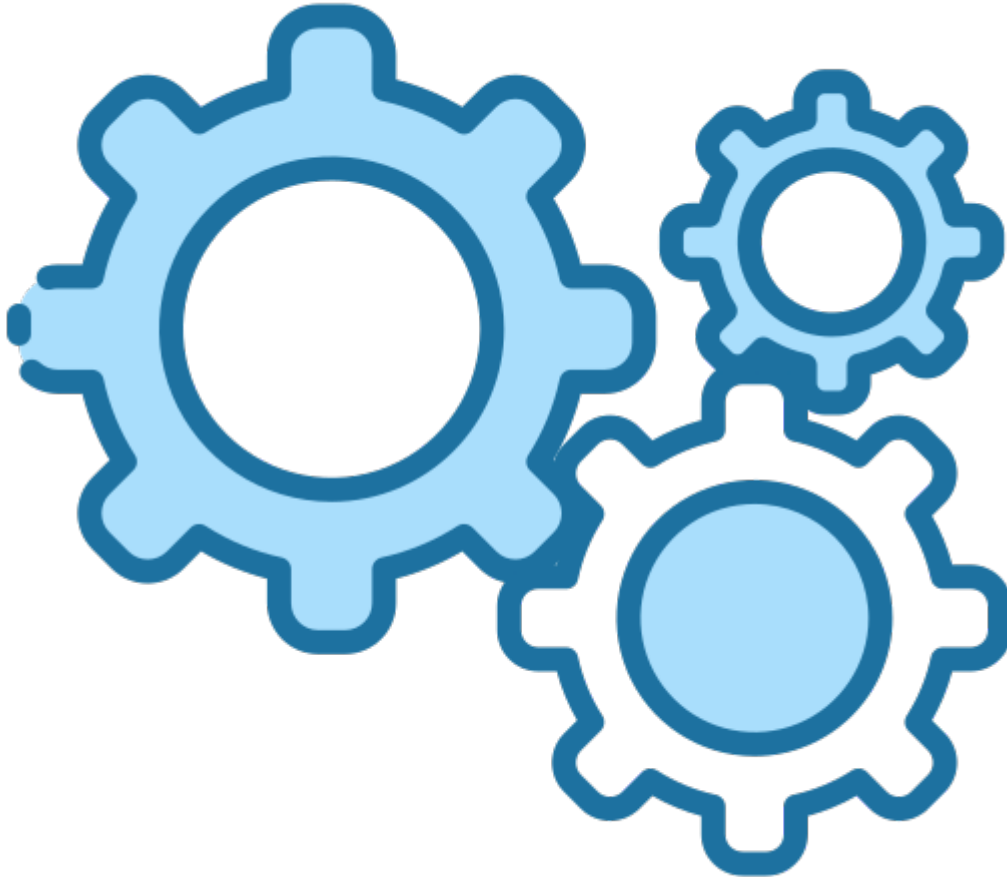
Security

All new accounts come automatically provisioned with AI monitoring of access attempts using *Allure* as well as monitoring for S3 buckets exposed to the public internet. *Lacework* is deployed to all Kubernetes clusters to detect anomalous access patterns.



Application Authentication Proxy

Patterns for implementing an authentication proxy are available so that you can secure the endpoints on your application with *Okta* without you writing any code.



AWS Integrations

The extra steps required for Amazon EKS to integrate with Amazon Elastic Block Store (Amazon EBS), Elastic Load Balancing (Application and Network Load Balancers) and AWS Identity and Access Management (IAM) are done for you so that you can get right to work.



Autoscaling

Your pods can make use of *KEDA* to enable easy pod autoscaling and node autoscaling (provided by *Karpenter* and *Compactor*, which automatically creates and removes nodes). Node autoscaling supports GPU and ARM node types.



Monitoring

Your clusters are monitored for conditions such as when `Pod s` are unable to schedule, IPs are about to run out, or node groups are running out of capacity.



Logging

Pod logging to *Loki* is deployed for you. Just login to Grafana to see the logs from all pods in your cluster.



Metrics

Metrics for your cluster are automatically scraped and sent to *Chronosphere* with easy dashboards pre-created in Grafana to give you a cluster overview.



Support

Support is provided on Slack ( [#cloud-ops-support-ext](#)) as well as access to an on-call engineer 24/7/365.

Architecture

The following architectural diagram is a visual representation of a Substrate environment when it is provisioned. It includes the hierarchy in the EPIC AWS Organization as it relates to the 2 accounts that are provisioned for each environment. It also shows the Kubernetes clusters that are created in each account and how each piece is interconnected and/or integrated with other AWS accounts and features that are provisioned in the environment.

Substrate Environment Architecture

Requesting a Substrate Environment

To request a new Substrate Environment, refer to the following: [Request a New Substrate Environment](#)

Page Information:

Page ID: 81068480

Space: Cloud Developer Platform

Downloaded: 2025-07-12 04:05:23