Customers run MinIO on Amazon Elastic Kubernetes Service (Amazon EKS) for the following three reasons.

01.

MinIO serves as a consistent storage layer in a hybrid-cloud or multi-cloud deployment scenario.

02.

MinIO is Kubernetes native and high performance it can deliver predictable performance across public, private and edge cloud environments.

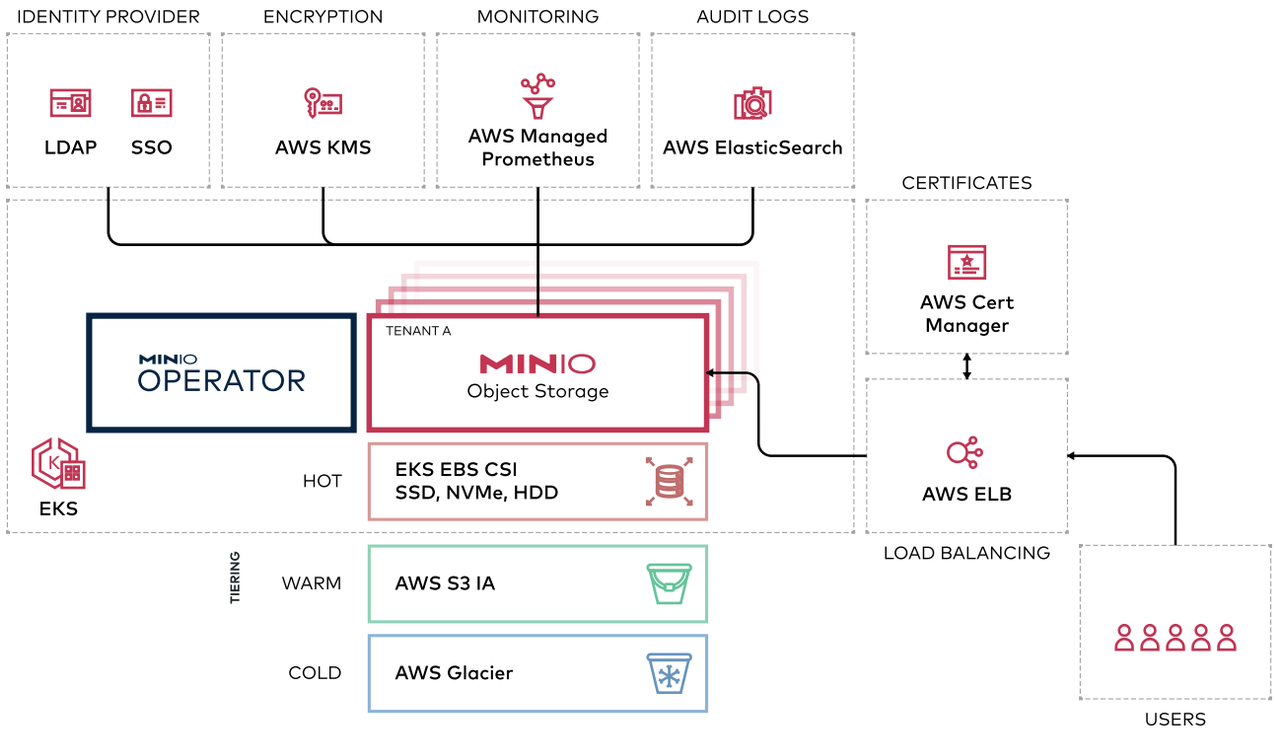
03.

Running MinIO on EKS provides control over the software stack with the attendant flexibility necessary to avoid cloud lock-in.

Architecture

Amazon Elastic Kubernetes Service (Amazon EKS) is a managed service that you can use to run Kubernetes on AWS without needing to install, operate, and maintain your own Kubernetes control plane or nodes.

MinIO provides a portable high-performance object storage system across all of the major Kubernetes platforms (Tanzu, Azure, GCP, OpenShift). On AWS, MinIO natively integrates with the Amazon EKS service making it easier to operate your own large scale multi-tenant object storage as a service. MinIO is a complete drop-in replacement for AWS S3 storage-as-a-service.



Unlike AWS S3, MinIO enables applications to scale across multi-cloud and hybrid cloud infrastructure, without requiring expensive software rewrites or proprietary integrations. Because MinIO is containerized and Kubernetes-native it can be rolled out across these platforms without requiring specialized skills to operate large scale storage infrastructure.

The MinIO Operator integrates natively with EKS and AWS features to provide:

Storage Classes and Tiering

Tier across AWS EBS, S3, S3 IA and Glacier.

The key requirement to deploy MinIO at scale on AWS is the ability tier across AWS storage classes (EBS, S3, S3 IA, S3 Glacier). This allows enterprises to manage both cost and performance.

MinIO can be configured to automatically transition aged objects from the hot EBS tier to cost-efficient warm and cold AWS S3 storage tiers. For example, MinIO tiering policy can be configured with EBS as the primary tier, S3 IA as the secondary tier and S3 Glacier as the tertiary or archival tier.

When tiering, MinIO maintains a single object storage namespace by transparently retrieving transitioned objects without additional client-side logic. MinIO also supports using the AWS restore API for bringing objects back onto the "hot" performance-optimized storage infrastructure.

This same capability also extends to hybrid cloud environments where the JBODs act as the hot tier on the private cloud side and S3 acts as the warm and cold tier. The data on the public cloud remains encrypted - ensuring that data is safe at rest and in flight.

Encryption Key Management

Manage encryption keys with AWS Key Management Service.

For cloud-native applications, the best practices dictate storing keys outside of the object system in an external vault. Amazon Key Management Service (KMS) is a secure and resilient service for managing keys by API across AWS services.

For those with more stringent security requirements or for consistency purposes, MinIO integrates with a number of external Key Management Services that operate outside of AWS.

When using MinIO as the object storage in the public EKS environment, encryption on-disk is strongly recommended. MinIO uses AES-256-GCM or ChaCha20-Poly1305 encryption to protect data integrity and confidentiality with negligible performance impact.

The MinIO Operator allows for tenants to be configured for the Amazon KMS or a supported third-party KMS for automatic server-side encryption of objects.

MinIO supports setting a bucket-level default encryption key in the KMS with support for AWS-S3 semantics (SSE-S3). Clients can also specify a separate key on the KMS using SSE-KMS request headers.

MinIO will use this KMS to bootstrap its internal key encryption server (KES service) to enable high-performance per object encryption. Each tenant runs its own KES server in an isolated

External Load Balancing

Load balance incoming requests with AWS Elastic Load Balancing.

The MinIO Operator integrates tightly with the AWS Elastic Load Balancer (ELB) to provide automatic load balancing and routing service across multiple MinIO tenants for applications accessing the storage service from outside of AWS. Exposing a MinIO tenant to external traffic can be done by simply adding annotations to a MinIO tenant. This service enables applications to scale and be accessed by hundreds of millions of devices across the planet.

Identity Management

Manage identity and policy with AWS Directory Service and AWS Identity Service.

When running MinIO on AWS EKS, customers can manage the single sign-on (SSO) through Amazon's hosted Identity Management Service or third party OpenID Connect/LDAP compatible identity providers like Keycloak, Okta/Auth0, Google, Facebook, ActiveDirectory and OpenLDAP.

A single, centralized IDP allows administrators to add, change privileges for, or eliminate a user, service account, or group once - and have it be enforced across all public cloud, private cloud and edge MinIO servers. The ability to have a unified identity and access management (IAM) layer independent of the infrastructure provides significant architectural flexibility.

Certificate Management

Configure and manage certificates with AWS Certificate Manager.

All traffic from the application to MinIO, including internode traffic, is encrypted with TLS. TLS certificates are used to secure network communications and establish the identity of network-connected resources, such as a MinIO server domain.

MinIO integrates with the AWS Certificate Manager to automatically configure, provision, manage and update certificates for the MinIO tenants. The tenants are completely isolated from each other in their own Kubernetes namespace with their own certificates for improved security.

Logging and Auditing

Output logs to AWS ElasticSearch for analysis.

Enabling MinIO auditing generates a log for every operation on the object storage cluster. In addition to the audit log, MinIO also logs console errors for operational troubleshooting purposes.

MinIO supports outputting logs to AWS Elasticsearch (or third parties) for analysis and alerting.

Monitoring and Alerting

Track metrics and issue alerts using AWS Managed Prometheus.

MinIO recommends using the Amazon Managed Service for Prometheus (AMP) for monitoring and alerting on MinIO EKS instances. MinIO publishes every object storage related Prometheus metric imaginable, from bucket capacity to access metrics. Those metrics can be collected and visualized in any Prometheus-compatible tool or the MinIO Console.

External monitoring solutions scrape the MinIO Prometheus endpoint at regular intervals. MinIO recommends either Grafana or the Amazon Managed Service for Prometheus (AMP) depending on the architectural goals. These same tools can also be used to establish baselines and set alert thresholds for notifications, which can then be routed to a notification platform such as PagerDuty, Freshservice or even SNMP.

Note: *MinIO does not recommend running MinIO in AWS if AWS is the only anticipated instance.*

Rather, MinIO in AWS should be reserved for scenarios where the organization seeks consistency across multiple environments.

MinIO offers the most deployment options of any object storage across hybrid, multicloud and edge environments using Kubernetes. Amazon Elastic Kubernetes Service users can quickly and easily deploy multi-tenant MinIO object storage within their existing environment.

The MinIO Kubernetes Operator supports end-to-end configuration and deployment of MinIO tenants, seamlessly integrating with supporting infrastructure for storage tiering, monitoring, audit logging, identity management, certificate management, key management for automatic server-side object encryption and load balancing and ingress.

Learn more at **min.io** or try it for yourself at **min.io/download**. Join the conversation at **minio.slack.com**.