Обзор

Архитектура

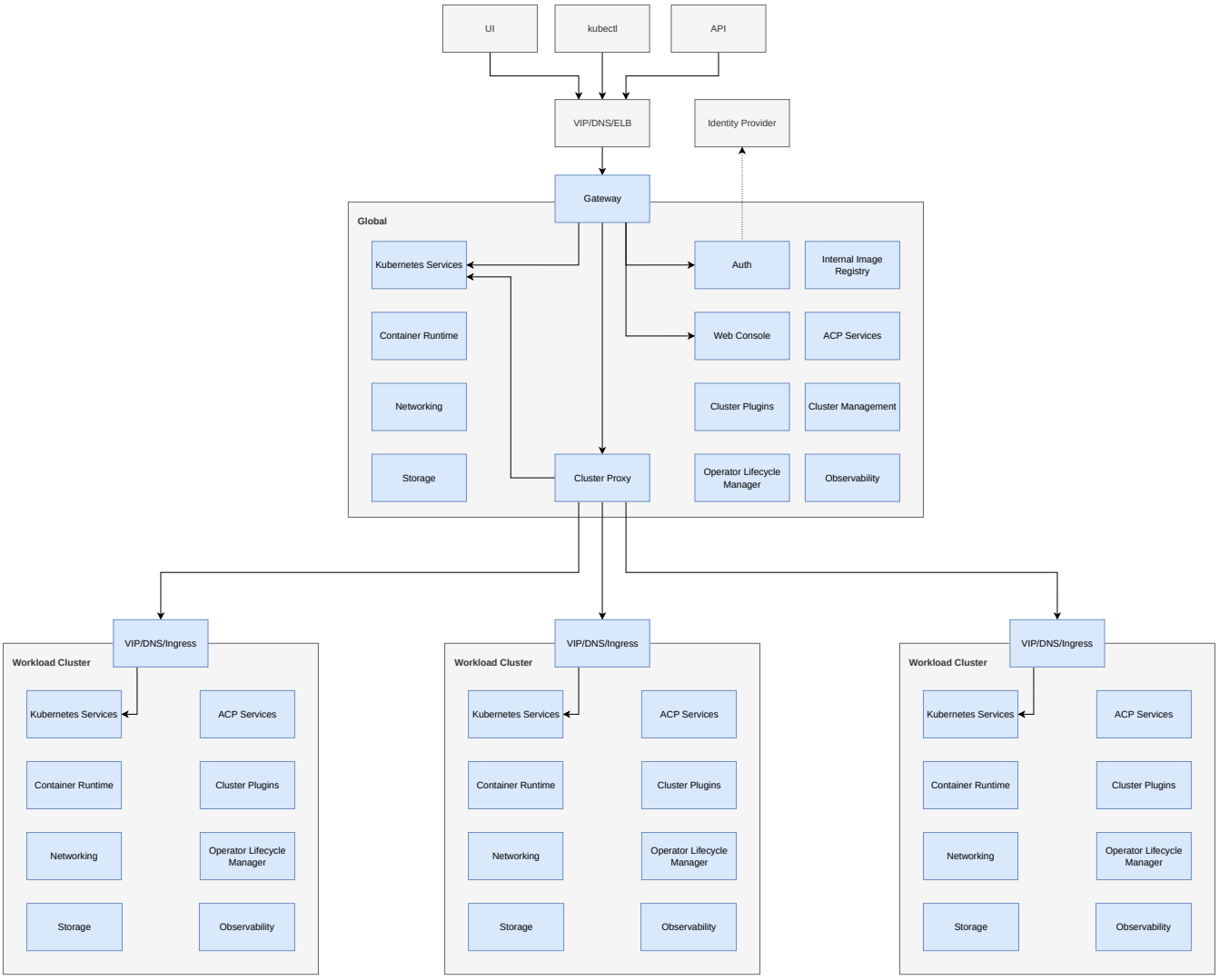
Содержание

Введение в Alauda Container Platform

Alauda Container Platform (ACP) предоставляет корпоративную платформу на базе Kubernetes, которая позволяет организациям создавать, развертывать и управлять приложениями последовательно в гибридных и мультиоблачных средах. ACP интегрирует базовые возможности Kubernetes с расширенными сервисами управления, наблюдаемости и безопасности, предлагая единый контрольный уровень и гибкие кластеры для рабочих нагрузок.

Архитектура построена по модели **hub-and-spoke**, состоящей из `global` кластера и множества кластеров рабочих нагрузок. Такая конструкция обеспечивает

централизованное управление при сохранении независимого выполнения и масштабируемости рабочих нагрузок.



Основные архитектурные компоненты

Глобальный кластер

`global` кластер служит централизованным узлом управления и контроля ACP. Он предоставляет платформенные сервисы, такие как аутентификация, управление политиками, операции жизненного цикла кластеров и наблюдаемость. Также это центральный узел для управления мультикластерной средой и обеспечивает функциональность между кластерами.

Ключевые компоненты включают:

- **Gateway**
  Выполняет роль основной точки входа в платформу. Управляет API-запросами от UI, CLI (kubectl) и инструментов автоматизации, маршрутизируя их к соответствующим backend-сервисам.
- **Аутентификация и авторизация (Auth)**
  Интегрируется с внешними Identity Providers (**IdPs**) для обеспечения единого входа (SSO) и контроля доступа на основе RBAC.
- **Веб-консоль**
  Предоставляет веб-интерфейс для ACP. Взаимодействует с API платформы через gateway.
- **Управление кластерами**
  Отвечает за регистрацию, подготовку и управление жизненным циклом кластеров рабочих нагрузок.
- **Сервисы ACP**
- **Operator Lifecycle Manager (OLM) и плагины кластера**
  Управляют установкой, обновлениями и жизненным циклом операторов и расширений кластера.
- **Внутренний реестр образов**
  Предоставляет встроенный репозиторий контейнерных образов с управлением доступом на основе ролей.
- **Наблюдаемость**
  Обеспечивает централизованный сбор логов, метрик и трассировки как для `global`, так и для рабочих кластеров.
- **Прокси кластера**
  Обеспечивает безопасную связь между `global` кластером и кластерами рабочих нагрузок.

Кластер рабочих нагрузок

Кластеры рабочих нагрузок — это Kubernetes-среды, управляемые `global` кластером. Каждый такой кластер запускает изолированные приложения и наследует управление и конфигурацию от центральной контрольной плоскости.

Внешние интеграции

- **Identity Provider (IdP)**
  Поддерживает федеративную аутентификацию через стандартные протоколы (OIDC, SAML) для унифицированного управления пользователями.
- **Доступ через API и CLI**
  Пользователи могут взаимодействовать с ACP через RESTful API, веб-консоль или командные инструменты, такие как `kubectl` и `ac`.
- **Балансировщик нагрузки (VIP/DNS/SLB)**
  Обеспечивает высокую доступность и распределение трафика для Gateway и ingress-точек доступа `global` и рабочих кластеров.

Масштабируемость и высокая доступность

ACP спроектирован для горизонтального масштабирования и высокой доступности:

- Каждый компонент может быть развернут с резервированием для устранения единой точки отказа.
- `global` кластер поддерживает управление десятками и сотнями кластеров рабочих нагрузок.
- Кластеры рабочих нагрузок могут масштабироваться независимо в зависимости от нагрузки.
- Использование VIP/DNS/Ingress обеспечивает бесшовное маршрутизирование и переключение при сбоях.

Функциональная перспектива

Полный функционал Alauda Container Platform (ACP) состоит из **ACP Core** и расширений, основанных на двух технических стеках: **Operator** и **Cluster Plugin**.

- **ACP Core**
  Минимальный поставляемый модуль ACP, предоставляющий основные возможности, такие как управление кластерами, оркестрация контейнеров, проекты и

администрирование пользователей.

- Соответствует самым высоким стандартам безопасности
- Обеспечивает максимальную стабильность
- Предлагает самый длительный жизненный цикл поддержки

- **Расширения**
  Расширения в стеках Operator и Cluster Plugin могут классифицироваться как:

  - **Aligned** – стратегия жизненного цикла с несколькими потоками поддержки, согласованная с ACP.
  - **Agnostic** – стратегия жизненного цикла с несколькими потоками поддержки, выпускаемая независимо от ACP.

  Подробнее о расширениях см. в разделе Extend.

Техническая перспектива

Выполнение компонентов платформы
Все компоненты платформы работают в контейнерах внутри кластера управления Kubernetes (`global` кластер).

Архитектура высокой доступности

- `global` кластер обычно состоит минимум из трёх узлов контрольной плоскости и нескольких рабочих узлов
- Высокая доступность etcd является ключевой для HA кластера; подробности см. в разделе *Key Component High Availability Mechanisms*
- Балансировка нагрузки может обеспечиваться внешним балансировщиком или собственным VIP внутри кластера

Маршрутизация запросов

- Клиентские запросы сначала проходят через балансировщик нагрузки или собственный VIP
- Запросы перенаправляются на **ALB** (стандартный Kubernetes Ingress Gateway платформы), работающий на выделенных ingress-узлах (или на узлах контрольной

плоскости, если настроено)

- ALB маршрутизирует трафик к целевым подам компонентов согласно настроенным правилам

Стратегия репликации

- Основные компоненты работают минимум с двумя репликами
- Ключевые компоненты (например, registry, MinIO, ALB) работают с тремя репликами

Отказоустойчивость и самовосстановление

- Достигается за счёт взаимодействия kubelet, kube-controller-manager, kube-scheduler, kube-proxy, ALB и других компонентов
- Включает проверки здоровья, переключение при сбоях и перенаправление трафика

Хранение данных и восстановление

- Конфигурация контрольной плоскости и состояние платформы хранятся в etcd как ресурсы Kubernetes
- При катастрофических сбоях восстановление возможно из снимков etcd

Основное / резервное аварийное восстановление

- Два отдельных `global` кластера: **Primary Cluster** и **Standby Cluster**
- Механизм аварийного восстановления основан на синхронизации данных etcd в реальном времени с Primary Cluster на Standby Cluster
- В случае недоступности Primary Cluster из-за сбоя сервисы могут быстро переключиться на Standby Cluster

Механизмы высокой доступности ключевых компонентов

etcd

- Развернут на трёх (или пяти) узлах контрольной плоскости
- Использует протокол RAFT для выбора лидера и репликации данных

- Развёртывания на трёх узлах выдерживают отказ одного узла; на пяти — двух узлов
- Поддерживает локальные и удалённые S3-снимки резервных копий

Компоненты мониторинга

- **Prometheus**: несколько инстансов, дедупликация с Thanos Query и кросс-региональная избыточность
- **VictoriaMetrics**: кластерный режим с распределёнными компонентами VMStorage, VMInsert и VMSelect

Компоненты логирования

- **Nevermore** собирает логи и данные аудита
- **Kafka / Elasticsearch / Razor / Lanaya** развернуты в распределённом режиме с несколькими репликами

Сетевые компоненты (CNI)

- **Kube-OVN / Calico / Flannel** достигают HA через stateless DaemonSets или трёхкратные реплики компонентов контрольной плоскости

ALB

- Оператор развернут с тремя репликами, включён выбор лидера
- Проверки здоровья на уровне инстансов и балансировка нагрузки

Собственный VIP

- Высокодоступный виртуальный IP на базе Keepalived
- Поддерживает обнаружение heartbeat и переключение active-standby

Harbor

- Балансировка нагрузки на базе ALB
- PostgreSQL с Patroni HA
- Redis в режиме Sentinel
- Статлес-сервисы развернуты с несколькими репликами

Registry и MinIO

- Registry развернут с тремя репликами
- MinIO в распределённом режиме с кодированием с удалением, избыточностью данных и автоматическим восстановлением

Kubernetes Support Matrix

Этот документ содержит матрицу поддержки версий Kubernetes для ACP. Эта информация важна при создании кластеров, обновлении ACP и управлении сторонними кластерами.

Содержание

Overview

ACP поддерживает несколько версий Kubernetes в разных выпусках ACP. Понимание поддерживаемых версий необходимо для:

- **Создания кластеров** – Определение, какие версии Kubernetes можно использовать при создании новых кластеров
- **Обновления ACP** – Обеспечение соответствия всех рабочих кластеров требованиям совместимости перед обновлением глобального кластера
- **Управления сторонними кластерами** – Проверка, что кластеры Kubernetes в публичных облаках или соответствующие CNCF находятся в поддерживаемом диапазоне версий

Version Support Matrix

В следующей таблице показана поддержка версий Kubernetes для каждого выпуска ACP.

INFO

В таблице указаны минорные версии ACP без различия патч-версий. Патч-версии содержат только исправления ошибок и обновления безопасности, поэтому минорные версии Kubernetes остаются одинаковыми во всех патч-версиях одного минорного выпуска.

Начиная с ACP 4.1, каждый выпуск ACP поддерживает только **одну версию Kubernetes** для создания кластера. Это обеспечивает согласованность и упрощает процесс обновления новых кластеров.

ACP Version	Supported for Cluster Creation	Compatible Versions
ACP 4.2	1.33	1.33, 1.32, 1.31, 1.30
ACP 4.1	1.32	1.32, 1.31, 1.30, 1.29
ACP 4.0	1.31, 1.30, 1.29, 1.28	1.31, 1.30, 1.29, 1.28

Upgrade Requirements

Для ACP 4.2 и более ранних версий **все** рабочие кластеры должны быть обновлены до **последней** версии Kubernetes из списка совместимых версий **до** обновления глобального кластера ACP.

В будущих выпусках рабочие кластеры будут должны находиться только в диапазоне совместимых версий для обновления глобального кластера ACP.

Примечания к выпуску

Содержание

4.2.3

Исправленные ошибки

- In version 4.2.2, the platform was unable to automatically synchronize data from the connected LDAP server. This issue has been fixed in version 4.2.3.
- In version 4.2.2, both creating project settings quotas and updating quotas for existing projects failed to take effect. This issue has been fixed in version 4.2.3.
- Fixed an issue where batch deletion operations on the container group list page failed due to "insufficient permissions". Restored missing RBAC permissions related to batch operations in the core management component.
- Enhanced Multus CNI security by enabling the service to run as a non-privileged user.
- Fix the deployment failure of the Argo Rollouts plugin in v4.2.2

Известные проблемы

- When the platform interfaces with IDP user authentication login, if the user name has capital letters, ArgoCD will not be able to get the permission information of the user. This issue has been resolved in version 4.2.4.

- When using Alauda Container Platform Monitoring for VictoriaMetrics with multiple clusters sharing the same Storage, the alert rule cpaas-certificates-rule has two issues: alert notifications do not differentiate between clusters when triggered, and the rule monitors customer secrets instead of only platform certificates.
- In certain cases, users may find that the operations automatically recorded by the platform in a ResourcePatch resource do not match the actual modifications they made to the component. As a result, when the ResourcePatch controller applies the patch, the component may undergo unexpected changes.
  Workaround: Users should manually modify the ResourcePatch resource to ensure it reflects the intended changes.
- When using violet push to upload a chart package, the push operation may complete successfully, but the package does not appear in the public-charts repository.
  Workaround: Push the chart package again.
- Although the installation page provides form fields for configuring labels and annotations for the global cluster, these configurations are not applied in practice.
- Fixed an issue where pushing images larger than 1GB would fail due to connection interruptions caused by the default 30-second timeout of the Registry Proxy. Extended the HTTP forward timeout of the Proxy component and enabled streaming to enhance transmission efficiency and stability for large data volumes.
- If a Custom Application includes an alert resource whose metrics expression uses customized metrics, deploying the application to a namespace whose name differs from the original—whether after exporting it as a chart or application YAML, importing the chart into the platform, or creating the application directly from the YAML — will cause the deployment to fail.

  This issue can be resolved by manually updating the metrics expression in the alert resource within the chart or YAML file, changing the namespace tag value to match the target deployment namespace.
- The default pool .mgr created by ceph-mgr uses the default Crush Rule, which may fail to properly select OSDs in a stretched cluster. To resolve this, the .mgr pool must be created using CephBlockPool. However, due to timing uncertainties, ceph-mgr might attempt to create the .mgr pool before the Rook Operator completes its setup, leading to conflicts.
  If encountering this issue, restart the rook-ceph-mgr Pod to trigger reinitialization.
  If unresolved, manually clean up the conflicting .mgr pool and redeploy the cluster to ensure proper creation order.

- Application creation failure triggered by the defaultMode field in YAML.
  Affected Path: Alauda Container Platform → Application Management → Application List → Create from YAML. Submitting YAML containing the defaultMode field (typically used for ConfigMap/Secret volume mount permissions) triggers validation errors and causes deployment failure.
  Workaround: Manually remove all defaultMode declarations before application creation.
- When pre-delete post-delete hook is set in helm chart.
  When the delete template application is executed and the chart is uninstalled, the hook execution fails for some reasons, thus the application cannot be deleted. It is necessary to investigate the cause and give priority to solving the problem of hook execution failure.

4.2.2

Исправленные ошибки

- Fixed etcd backup failure after installing Alauda Container Platform Cluster Enhancer plugin. The issue was caused by BroadcastJobs created during upgrade missing the scheduledTimeAnnotation, preventing the AdvancedCronJob from calculating the next trigger time. The fix now uses the job creation timestamp as a fallback when the annotation is missing, ensuring scheduled backups continue to run properly.Fixed in ACP 4.2.2.
- Fixed an issue where the olm-registry pod would continuously restart, preventing the OperatorHub from functioning properly. This was caused by the `seccompProfile: RuntimeDefault` security configuration added during CIS compliance hardening, which blocked the `clone` syscall required by CGO operations. The seccomp profile has been adjusted to allow necessary syscalls while maintaining security compliance.Fixed in ACP 4.2.2.
- Fixed a performance issue where the permission validation during native application creation became extremely slow (10+ seconds) when the cluster had 60+ operators installed.Fixed in ACP 4.2.2.
- Resolved an issue where the system failed to retrieve storage device DeviceIDs, which caused the web console to incorrectly display the device status as "Not Recommended."
- Fix the issue where the egress gateway cannot route traffic from Pods in the same subnet as the egress gateway.

Известные проблемы

- When the platform interfaces with IDP user authentication login, if the user name has capital letters, ArgoCD will not be able to get the permission information of the user. This issue has been resolved in version 4.2.4.
- In version 4.2.2, the platform was unable to automatically synchronize data from the connected LDAP server. This issue has been fixed in version 4.2.3.
- In version 4.2.2, both creating project settings quotas and updating quotas for existing projects failed to take effect. This issue has been fixed in version 4.2.3.
- When using Alauda Container Platform Monitoring for VictoriaMetrics with multiple clusters sharing the same Storage, the alert rule cpaas-certificates-rule has two issues: alert notifications do not differentiate between clusters when triggered, and the rule monitors customer secrets instead of only platform certificates.
- In certain cases, users may find that the operations automatically recorded by the platform in a ResourcePatch resource do not match the actual modifications they made to the component. As a result, when the ResourcePatch controller applies the patch, the component may undergo unexpected changes.
  Workaround: Users should manually modify the ResourcePatch resource to ensure it reflects the intended changes.
- When using violet push to upload a chart package, the push operation may complete successfully, but the package does not appear in the public-charts repository.
  Workaround: Push the chart package again.
- Although the installation page provides form fields for configuring labels and annotations for the global cluster, these configurations are not applied in practice.
- Fixed an issue where pushing images larger than 1GB would fail due to connection interruptions caused by the default 30-second timeout of the Registry Proxy. Extended the HTTP forward timeout of the Proxy component and enabled streaming to enhance transmission efficiency and stability for large data volumes.
- Fixed an issue where batch deletion operations on the container group list page failed due to "insufficient permissions". Restored missing RBAC permissions related to batch operations in the core management component.
- Fix the deployment failure of the Argo Rollouts plugin in v4.2.2
- If a Custom Application includes an alert resource whose metrics expression uses customized metrics, deploying the application to a namespace whose name differs from the

original—whether after exporting it as a chart or application YAML, importing the chart into the platform, or creating the application directly from the YAML — will cause the deployment to fail.

This issue can be resolved by manually updating the metrics expression in the alert resource within the chart or YAML file, changing the namespace tag value to match the target deployment namespace.

- The default pool .mgr created by ceph-mgr uses the default Crush Rule, which may fail to properly select OSDs in a stretched cluster. To resolve this, the .mgr pool must be created using CephBlockPool. However, due to timing uncertainties, ceph-mgr might attempt to create the .mgr pool before the Rook Operator completes its setup, leading to conflicts.
  If encountering this issue, restart the rook-ceph-mgr Pod to trigger reinitialization.
  If unresolved, manually clean up the conflicting .mgr pool and redeploy the cluster to ensure proper creation order.
- Application creation failure triggered by the defaultMode field in YAML.
  Affected Path: Alauda Container Platform → Application Management → Application List → Create from YAML. Submitting YAML containing the defaultMode field (typically used for ConfigMap/Secret volume mount permissions) triggers validation errors and causes deployment failure.
  Workaround: Manually remove all defaultMode declarations before application creation.
- When pre-delete post-delete hook is set in helm chart.
  When the delete template application is executed and the chart is uninstalled, the hook execution fails for some reasons, thus the application cannot be deleted. It is necessary to investigate the cause and give priority to solving the problem of hook execution failure.

4.2.1

Исправленные ошибки

- When upgrading the Global cluster, there was an intermittent issue that caused the Marketplace menu to be missing from the left navigation in the Web Console. This issue has been fixed in ACP 4.2.1.
- When using the etcd backup feature provided by Alauda Container Platform Cluster Enhancer, if users configure to back up etcd to S3 storage, the plugin fails to retrieve the

Secret object referenced in secretRef. The root cause was that the plugin lacked the necessary RBAC permissions to read Secrets, resulting in S3 authentication information retrieval failure. This issue has been fixed in ACP 4.2.1.

- When using the Global Cluster Disaster Recovery solution, after installing Alauda Container Platform etcd Synchronizer, the Web Console of the Standby Cluster was unable to login. The root cause was that etcd Synchronizer did not properly ignore k8sadmin-* related Secret objects, causing the authentication information of the Standby Cluster to be overwritten. This issue has been fixed in ACP 4.2.1.
- Fixed an issue where HTTPS certificate verification could not be skipped when connecting to MinIO.

Известные проблемы

- When the platform interfaces with IDP user authentication login, if the user name has capital letters, ArgoCD will not be able to get the permission information of the user. This issue has been resolved in version 4.2.4.
- Fixed etcd backup failure after installing Alauda Container Platform Cluster Enhancer plugin. The issue was caused by BroadcastJobs created during upgrade missing the scheduledTimeAnnotation, preventing the AdvancedCronJob from calculating the next trigger time. The fix now uses the job creation timestamp as a fallback when the annotation is missing, ensuring scheduled backups continue to run properly.Fixed in ACP 4.2.2.
- Fixed an issue where the olm-registry pod would continuously restart, preventing the OperatorHub from functioning properly. This was caused by the `seccompProfile: RuntimeDefault` security configuration added during CIS compliance hardening, which blocked the `clone` syscall required by CGO operations. The seccomp profile has been adjusted to allow necessary syscalls while maintaining security compliance.Fixed in ACP 4.2.2.
- Fixed a performance issue where the permission validation during native application creation became extremely slow (10+ seconds) when the cluster had 60+ operators installed.Fixed in ACP 4.2.2.
- When using Alauda Container Platform Monitoring for VictoriaMetrics with multiple clusters sharing the same Storage, the alert rule cpaas-certificates-rule has two issues: alert notifications do not differentiate between clusters when triggered, and the rule monitors customer secrets instead of only platform certificates.

- In certain cases, users may find that the operations automatically recorded by the platform in a ResourcePatch resource do not match the actual modifications they made to the component. As a result, when the ResourcePatch controller applies the patch, the component may undergo unexpected changes.
  Workaround: Users should manually modify the ResourcePatch resource to ensure it reflects the intended changes.
- When using violet push to upload a chart package, the push operation may complete successfully, but the package does not appear in the public-charts repository.
  Workaround: Push the chart package again.
- Although the installation page provides form fields for configuring labels and annotations for the global cluster, these configurations are not applied in practice.
- Resolved an issue where the system failed to retrieve storage device DeviceIDs, which caused the web console to incorrectly display the device status as "Not Recommended."
- Fix the issue where the egress gateway cannot route traffic from Pods in the same subnet as the egress gateway.
- If a Custom Application includes an alert resource whose metrics expression uses customized metrics, deploying the application to a namespace whose name differs from the original—whether after exporting it as a chart or application YAML, importing the chart into the platform, or creating the application directly from the YAML — will cause the deployment to fail.

  This issue can be resolved by manually updating the metrics expression in the alert resource within the chart or YAML file, changing the namespace tag value to match the target deployment namespace.
- The default pool .mgr created by ceph-mgr uses the default Crush Rule, which may fail to properly select OSDs in a stretched cluster. To resolve this, the .mgr pool must be created using CephBlockPool. However, due to timing uncertainties, ceph-mgr might attempt to create the .mgr pool before the Rook Operator completes its setup, leading to conflicts.
  If encountering this issue, restart the rook-ceph-mgr Pod to trigger reinitialization.
  If unresolved, manually clean up the conflicting .mgr pool and redeploy the cluster to ensure proper creation order.
- Application creation failure triggered by the defaultMode field in YAML.
  Affected Path: Alauda Container Platform → Application Management → Application List → Create from YAML. Submitting YAML containing the defaultMode field (typically used for ConfigMap/Secret volume mount permissions) triggers validation errors and causes

deployment failure.
Workaround: Manually remove all defaultMode declarations before application creation.

- When pre-delete post-delete hook is set in helm chart.
  When the delete template application is executed and the chart is uninstalled, the hook execution fails for some reasons, thus the application cannot be deleted. It is necessary to investigate the cause and give priority to solving the problem of hook execution failure.

4.2.0

Новые возможности и улучшения

Поддержка Kubernetes 1.33

ACP теперь поддерживает **Kubernetes 1.33**, предоставляя последние функции, улучшения производительности и повышения безопасности от сообщества Kubernetes.

ACP CLI (ac)

Новый **ACP CLI (ac)** позволяет разрабатывать, собирать, развёртывать и запускать приложения на ACP с удобным интерфейсом командной строки.

Основные возможности включают:

- **Команды, совместимые с kubectl**
- **Интегрированная аутентификация** с окружениями платформы ACP
- **Единое управление сессиями** между несколькими окружениями
- **Расширения, специфичные для ACP**, для доступа к платформе и кросс-окруженческих рабочих процессов

Для полного описания возможностей смотрите: ACP CLI (ac)

Hosted Control Plane (HCP)

Выпущено:

- **Alauda Container Platform Kubeadm Provider**
- **Alauda Container Platform Hosted Control Plane**
- **Alauda Container Platform SSH Infrastructure Provider**

Жизненный цикл: *Agnostic* (выпускается асинхронно с ACP)

Hosted Control Plane отделяет управляющую плоскость от рабочих узлов, размещая управляющие компоненты каждого кластера в контейнерах внутри управляющего кластера. Такая архитектура снижает потребление ресурсов, ускоряет создание и обновление кластеров, а также обеспечивает лучшую масштабируемость для больших мультикластерных сред.

Подробнее см.: About Hosted Control Plane

Улучшенное управление правами пользователей

Мы оптимизировали управление RBAC с помощью следующих улучшений для повышения удобства и поддержки:

- **Управление ролями платформы:**
  - **Настройка прав через UI устарела:** Роли платформы больше не поддерживают настройку прав через веб-консоль. Все права ролей должны настраиваться через YAML-файлы.
  - **Обратная совместимость сохранена:** Существующие предустановленные роли платформы и роли, определённые в предыдущих версиях, остаются полностью функциональными. Пользователи могут продолжать назначать эти роли и предоставлять права как раньше.
- **Управление ролями Kubernetes:**
  - **Нативное управление ролями Kubernetes:** В консоли платформы теперь доступен отдельный интерфейс управления ресурсами Role и ClusterRole Kubernetes, позволяющий напрямую связывать Kubernetes роли с пользователями и назначать права.
  - **Модульное определение прав:** Права ресурсов плагинов платформы будут постепенно мигрированы в отдельные ресурсы Role и ClusterRole, обеспечивая лучшую изоляцию и удобство управления.

Расширенные политики безопасности Pod с Kyverno

Мы усилили возможности безопасности рабочих нагрузок с помощью движка политик Kyverno:

- Готовые к использованию шаблоны безопасности: 8 проверенных шаблонов политик безопасности, встроенных в консоль, охватывающих уровни Pod Security Standards, включая Privileged, Baseline, Nonroot и Restricted
- Конфигурация в один клик: Быстро создавайте политики из шаблонов в бизнес-виде без ручного написания YAML, с немедленным применением в указанных пространствах имён

Следующее поколение Gateway API на базе Envoy Gateway

В этом выпуске представлена новая реализация Gateway API на основе Envoy Gateway. Она обеспечивает единый входящий трафик уровня L7, соответствует спецификации Gateway API сообщества и закладывает основу для более богатых политик трафика и интеграций экосистемы.

Правила на основе доменов для Egress Firewall

Egress Firewall теперь поддерживает правила разрешения/запрета на основе доменных имён, а не только IP-адресов. Это позволяет более тонко контролировать исходящий доступ к публичным SaaS-сервисам и внешним ресурсам с часто меняющимися IP-адресами.

Новый Endpoint Health Checker для более быстрого переключения

Введён новый Endpoint Health Checker, который быстрее обнаруживает сбои, такие как падения узлов и сетевые разрывы, и своевременно удаляет неисправные бэкенды. Это значительно сокращает время переключения трафика и снижает риск прерывания сервиса.

Новый оператор Local Storage для упрощённого управления Ceph/TopoLVM

Новый оператор **Local Storage** (Alauda Build of Local Storage) значительно упрощает развертывание и управление дисками для Ceph и TopoLVM. При развертывании можно просмотреть список всех доступных дисков в кластере с указанием модели, ёмкости и других ключевых атрибутов, а также выбрать диски для управления. Для привязки дисков Ceph и TopoLVM теперь предпочитают использовать device ID вместо путей монтирования, что предотвращает проблемы с хранилищем при изменении имён устройств после перезагрузки узла или повторного обнаружения устройств.

Другие ключевые изменения

Изменение жизненного цикла плагинов логирования

Статус жизненного цикла плагинов, связанных с логированием, изменён с *Aligned* на *Agnostic* (выпускаются асинхронно с ACP).

Затронутые плагины:

- **Alauda Container Platform Log Essentials** (новинка в этом выпуске)
- **Alauda Container Platform Log Storage for ClickHouse**
- **Alauda Container Platform Log Storage for Elasticsearch**
- **Alauda Container Platform Log Collector**

Подробнее см.: About Logging Service

Повышенный уровень безопасности по умолчанию для пространств имён

Начиная с версии 4.2.0, политика PSA по умолчанию для вновь создаваемых пространств имён (через веб-консоль или CLI) изменена с Baseline на Restricted.

- Baseline: Запрещает известные эскалации привилегий, обеспечивает умеренный уровень безопасности
- Restricted: Следует лучшим практикам безопасности Pod с самыми строгими требованиями

WARNING

Политика Restricted накладывает очень строгие требования к конфигурации Pod. Если вашему бизнесу необходимы возможности, такие как привилегированный режим, запуск от root-пользователя, монтирование путей хоста или использование сети хоста, такие рабочие нагрузки не смогут запускаться в пространствах имён с политикой Restricted по умолчанию.

Анализ влияния:

- Изменение затрагивает только вновь создаваемые пространства имён
- Рабочие нагрузки, требующие привилегий (например, root, hostPath), не будут запускаться напрямую

Рекомендуемые решения:

- Измените конфигурацию приложения, чтобы соответствовать требованиям политики Restricted
- При необходимости вручную установите политику пространства имён обратно на Baseline

MinIO в режиме обслуживания

MinIO (Alauda Build of MinIO) переведён в **режим обслуживания**. В дальнейшем будут выпускаться только исправления безопасности, новых функций не планируется. Существующие кластеры MinIO могут продолжать работу, а для новых требований к объектному хранилищу рекомендуется использовать Ceph Object.

Calico в режиме обслуживания

Плагин CNI **Calico** (Alauda Container Platform Networking for Calico) переведён в **режим обслуживания**. Мы будем устранять только вопросы безопасности, и он больше не является рекомендуемым сетевым решением по умолчанию. Существующие кластеры Calico остаются поддерживаемыми, а для новых кластеров рекомендуется использовать kube-ovn как стандартный CNI.

Ingress Nginx переведён на Operator

Ingress Nginx (Alauda Build of Ingress Nginx) мигрировал с плагина кластера на модель развертывания и управления через Operator. Существующие ресурсы Ingress продолжат

работать после обновления, а последующие операции предполагается выполнять через Operator. Несмотря на то, что версия Ingress Nginx от сообщества больше не обновляется, мы продолжим выпускать исправления ошибок и патчи безопасности для этой сборки.

Устаревшие и удалённые функции

Обновление политики обновления версии Kubernetes

Начиная с **ACP 4.2**, обновление версии Kubernetes **больше не является опциональным**. При обновлении кластера версия Kubernetes должна обновляться вместе с другими компонентами платформы. Это изменение обеспечивает согласованность версий в кластере и сокращает окна обслуживания в будущем.

ALB устарел начиная с v4.2.0

ALB (Alauda Container Platform Ingress Gateway) объявлен устаревшим с версии v4.2.0. Новые кластеры и пользователи должны использовать Gateway API на базе Envoy Gateway в качестве основного варианта. Существующие кластеры с ALB продолжат работу после обновления, но настоятельно рекомендуется планировать и проводить миграцию на Gateway API для долгосрочной поддержки и развития функционала.

Flannel полностью удалён

Плагин CNI **Flannel** (Alauda Container Platform Networking for Flannel) полностью удалён из платформы. Кластеры, которые всё ещё используют Flannel, должны мигрировать на kube-ovn до обновления до этого или более позднего релиза. Пожалуйста, планируйте и завершайте миграцию заранее, чтобы избежать прерывания сервиса из-за смены CNI.

Исправленные ошибки

- Previously, the status field of an upmachinepool resource stored the associated machine resources without any ordering. This caused the resource to be updated on every reconcile loop, resulting in excessively large audit logs. This issue has now been fixed.
- When the platform has a large number of clusters, the project quota of a single cluster cannot be updated after the quota is set for the project using the batch set project quota

function. This issue has been fixed.

- Previously, when creating a cluster-level Instance in OperatorHub, the web console automatically injected a metadata.namespace field, which caused a 404 error. This issue has been fixed in ACP 4.2.0.
- When a user is automatically disabled by the system due to long-term lack of login, it will be automatically disabled again after being manually activated by the administrator. This issue has been fixed.
- Previously, after uninstalling an Operator, the Operator status was incorrectly displayed as Absent, even though the Operator was actually Ready. Users had to manually re-upload the Operator using violet upload. This issue has now been resolved, and the Operator correctly appears as Ready after uninstallation.
- In some cases, installing a new Operator version after uploading it via violet upload would fail unexpectedly. This intermittent issue has been fixed.
- When an Operator or Cluster Plugin included multiple frontend extensions, the left-side navigation of these extensions could become unresponsive. The temporary workaround required users to add the annotation cpaas.io/auto-sync: "false" to the extension’s ConfigMap. This behavior has now been permanently fixed in the code.
- Previously, if a cluster contained nodes with an empty Display Name, users were unable to filter nodes by typing in the node selector dropdown on the node details page. This issue has been fixed in ACP 4.2.0.
- The temporary files were not deleted after log archiving, preventing disk space from being reclaimed. This issue has been fixed.
- Uploading multiple packages from a folder using violet upload previously failed when disk space became insufficient. Violet now proactively cleans up uploaded packages in time, preventing these errors.
- Fixed an issue where modifying the Pod Security Policy when importing a namespace into a project did not take effect.
- Fixed an issue where the monitoring dashboards for workloads (e.g., Applications, Deployments) in Workload clusters failed to display when the global cluster was upgraded while the Workload clusters remained un-upgraded.
- Fixed an issue causing the KubeVirt Operator deployment to fail when upgrading on Kubernetes versions prior to 1.30.

- Fixed an inconsistency where Secrets created through the web console only stored the Username and Password, and lacked the complete authentication field (auth) compared to those created via kubectl create. This issue previously caused authentication failures for build tools (e.g., buildah) that rely on the complete auth data.

Известные проблемы

- When using Alauda Container Platform Monitoring for VictoriaMetrics with multiple clusters sharing the same Storage, the alert rule cpaas-certificates-rule has two issues: alert notifications do not differentiate between clusters when triggered, and the rule monitors customer secrets instead of only platform certificates.
- When upgrading the Global cluster, there was an intermittent issue that caused the Marketplace menu to be missing from the left navigation in the Web Console. This issue has been fixed in ACP 4.2.1.
- When using the etcd backup feature provided by Alauda Container Platform Cluster Enhancer, if users configure to back up etcd to S3 storage, the plugin fails to retrieve the Secret object referenced in secretRef. The root cause was that the plugin lacked the necessary RBAC permissions to read Secrets, resulting in S3 authentication information retrieval failure. This issue has been fixed in ACP 4.2.1.
- When using the Global Cluster Disaster Recovery solution, after installing Alauda Container Platform etcd Synchronizer, the Web Console of the Standby Cluster was unable to login. The root cause was that etcd Synchronizer did not properly ignore k8sadmin-* related Secret objects, causing the authentication information of the Standby Cluster to be overwritten. This issue has been fixed in ACP 4.2.1.
- When using violet push to upload a chart package, the push operation may complete successfully, but the package does not appear in the public-charts repository.
  Workaround: Push the chart package again.
- Although the installation page provides form fields for configuring labels and annotations for the global cluster, these configurations are not applied in practice.
- Fixed an issue where HTTPS certificate verification could not be skipped when connecting to MinIO.
- If a Custom Application includes an alert resource whose metrics expression uses customized metrics, deploying the application to a namespace whose name differs from the original—whether after exporting it as a chart or application YAML, importing the chart into

the platform, or creating the application directly from the YAML — will cause the deployment to fail.

This issue can be resolved by manually updating the metrics expression in the alert resource within the chart or YAML file, changing the namespace tag value to match the target deployment namespace.

- The default pool .mgr created by ceph-mgr uses the default Crush Rule, which may fail to properly select OSDs in a stretched cluster. To resolve this, the .mgr pool must be created using CephBlockPool. However, due to timing uncertainties, ceph-mgr might attempt to create the .mgr pool before the Rook Operator completes its setup, leading to conflicts.
  If encountering this issue, restart the rook-ceph-mgr Pod to trigger reinitialization.
  If unresolved, manually clean up the conflicting .mgr pool and redeploy the cluster to ensure proper creation order.
- Application creation failure triggered by the defaultMode field in YAML.
  Affected Path: Alauda Container Platform → Application Management → Application List → Create from YAML. Submitting YAML containing the defaultMode field (typically used for ConfigMap/Secret volume mount permissions) triggers validation errors and causes deployment failure.
  Workaround: Manually remove all defaultMode declarations before application creation.
- When pre-delete post-delete hook is set in helm chart.
  When the delete template application is executed and the chart is uninstalled, the hook execution fails for some reasons, thus the application cannot be deleted. It is necessary to investigate the cause and give priority to solving the problem of hook execution failure.