

@serve.zone/cloudly

an Open Source solution for workload management at Enterprise scale.

- [readme.md for @serve.zone/cloudly](#)
- [changelog.md for @serve.zone/cloudly](#)

readme.md for @serve.zone/cloudly

Multi-cloud orchestration made simple. Manage containerized applications across cloud providers with Docker Swarmkit, featuring web dashboards, CLI tools, and powerful APIs.

☐ What is Cloudly?

Cloudly is your command center for multi-cloud infrastructure. It abstracts away the complexity of managing resources across different cloud providers while giving you the power and flexibility you need for modern DevOps workflows.

☐ Key Features

- ☐ **Multi-Cloud Management** - Seamlessly orchestrate resources across Cloudflare, Hetzner, DigitalOcean and more
- ☐ **Docker Swarmkit Integration** - Native container orchestration with production-grade reliability
- ☐ **Secret Management** - Secure handling of credentials, API keys, and sensitive configuration
- ☐ **Web Dashboard** - Beautiful, responsive UI built with modern web components
- ☐ **CLI & API** - Full programmatic control through TypeScript/JavaScript APIs and command-line tools
- ☐ **SSL/TLS Automation** - Automatic certificate provisioning via Let's Encrypt
- ☐ **Comprehensive Logging** - Built-in log aggregation and monitoring capabilities
- ☐ **Task Scheduling** - Automated workflows and background job management

☐ Quick Start

Installation

```
# Install the main package
pnpm add @serve.zone/cloudly

# Or install the CLI globally
pnpm add -g @serve.zone/cli

# Or just the API client
pnpm add @serve.zone/api
```

Basic Setup

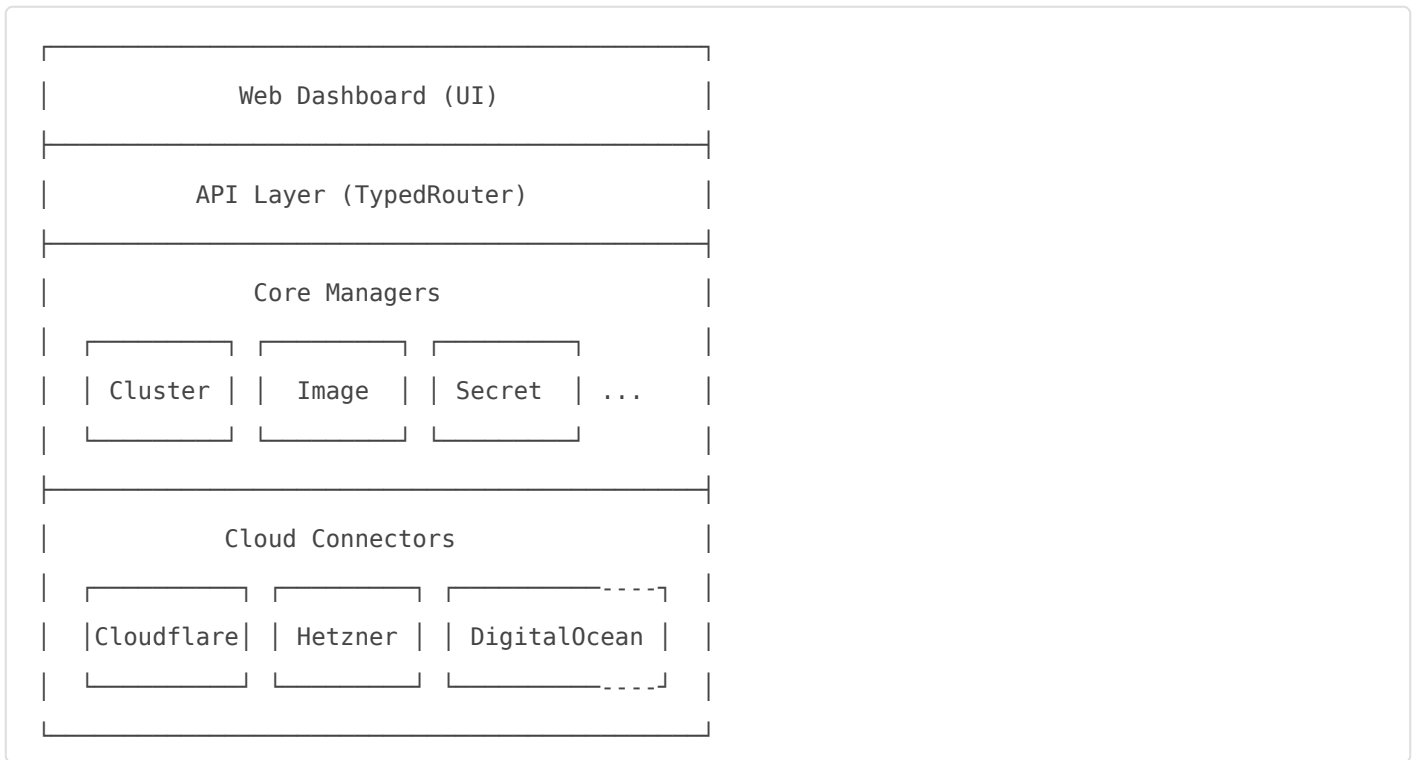
```
import { Cloudly } from '@serve.zone/cloudly';

// Initialize Cloudly with your configuration
const cloudly = new Cloudly({
  cfToken: process.env.CLOUDFLARE_TOKEN,
  hetznerToken: process.env.HETZNER_TOKEN,
  environment: 'production',
  letsEncryptEmail: 'certs@example.com',
  publicUrl: 'cloudly.example.com',
  publicPort: 443,
  mongoDescriptor: {
    mongoDbUrl: process.env.MONGODB_URL,
    mongoDbName: 'cloudly',
    mongoDbUser: process.env.MONGODB_USER,
    mongoDbPass: process.env.MONGODB_PASS,
  }
});

// Start the platform
await cloudly.start();
console.log('☑️ Cloudly is running!');
```

☑️ Architecture

Cloudly follows a modular architecture with clear separation of concerns:



☐☐ Core Components

☐☐ Managers

- **AuthManager** - Identity and access management
- **ClusterManager** - Docker Swarm cluster orchestration
- **ImageManager** - Container image lifecycle management
- **SecretManager** - Secure credential storage and distribution
- **ServerManager** - Cloud server provisioning and management
- **TaskManager** - Background job scheduling and execution

☐☐ Connectors

- **CloudflareConnector** - DNS, CDN, and edge services
- **LetsencryptConnector** - Automatic SSL certificate provisioning
- **MongodbConnector** - Database persistence layer
- **HetznerConnector** - German cloud infrastructure
- **DigitalOceanConnector** - Developer-friendly cloud resources

☐☐ Usage Examples

Managing Clusters

```
// Create a new cluster
const cluster = await cloudly.clusterManager.createCluster({
  name: 'production-cluster',
  region: 'eu-central',
  nodeCount: 3
});

// Deploy a service
await cluster.deployService({
  name: 'api-service',
  image: 'myapp:latest',
  replicas: 3,
  ports: [{ published: 80, target: 3000 }]
});
```

Secret Management

```
// Create a secret group
const secretGroup = await cloudly.secretManager.createSecretGroup({
  name: 'api-credentials',
  secrets: [
    { key: 'API_KEY', value: process.env.API_KEY },
    { key: 'DB_PASSWORD', value: process.env.DB_PASSWORD }
  ]
});

// Create a bundle for deployment
const bundle = await cloudly.secretManager.createSecretBundle({
  name: 'production-secrets',
  secretGroups: [secretGroup]
});
```

DNS Management

```
// Create DNS records via Cloudflare
const record = await cloudly.cloudflareConnector.createDNSRecord(
  'example.com',
  'api.example.com',
  'A',
  '192.168.1.1'
);
```

Web Dashboard

```
import { html } from '@design.estate/dees-element';

// Create a custom dashboard view
const dashboard = html`
  <cloudly-dashboard>
    <cloudly-view-clusters></cloudly-view-clusters>
    <cloudly-view-dns></cloudly-view-dns>
    <cloudly-view-images></cloudly-view-images>
  </cloudly-dashboard>
`;

document.body.appendChild(dashboard);
```

CLI Usage

The CLI provides quick access to all Cloudly features:

```
# Login to your Cloudly instance
servezone login --url https://cloudly.example.com

# List clusters
servezone clusters list

# Deploy a service
servezone deploy --cluster prod --image myapp:latest
```

```
# Manage secrets
servezone secrets create --name api-key --value "secret123"

# View logs
servezone logs --service api-service --follow
```

☐☐ Package Exports

This monorepo publishes multiple packages:

- **@serve.zone/cloudly** - Main orchestration platform
- **@serve.zone/api** - TypeScript/JavaScript API client
- **@serve.zone/cli** - Command-line interface
- **@serve.zone/interfaces** - Shared TypeScript interfaces

☐☐ Security Features

- **End-to-end encryption** for secrets
- **Role-based access control** (RBAC)
- **Automatic SSL/TLS** certificate management
- **Secure token-based authentication**
- **Audit logging** for compliance

☐☐ Production Ready

Cloudly is battle-tested in production environments managing:

- High-traffic web applications
- Microservice architectures
- CI/CD pipelines
- Data processing workloads
- Real-time communication systems

☐☐ Development

```
# Clone the repository
git clone https://gitlab.com/servezone/private/cloudly.git

# Install dependencies
pnpm install

# Run tests
pnpm test

# Build the project
pnpm build

# Start development mode
pnpm watch
```

Documentation

For detailed documentation, API references, and guides, visit our [documentation site](#).

License and Legal Information

This repository contains open-source code that is licensed under the MIT License. A copy of the MIT License can be found in the [license](#) file within this repository.

Please note: The MIT License does not grant permission to use the trade names, trademarks, service marks, or product names of the project, except as required for reasonable and customary use in describing the origin of the work and reproducing the content of the NOTICE file.

Trademarks

This project is owned and maintained by Task Venture Capital GmbH. The names and logos associated with Task Venture Capital GmbH and any related products or services are trademarks of Task Venture Capital GmbH and are not included within the scope of the MIT license granted herein. Use of these trademarks must comply with Task Venture Capital GmbH's Trademark Guidelines, and any usage must be approved in writing by Task Venture Capital GmbH.

Company Information

Task Venture Capital GmbH

Registered at District court Bremen HRB 35230 HB, Germany

For any legal inquiries or if you require further information, please contact us via email at hello@task.vc.

By using this repository, you acknowledge that you have read this section, agree to comply with its terms, and understand that the licensing of the code does not imply endorsement by Task Venture Capital GmbH of any derivative works.

changelog.md for @serve.zone/cloudly

2025-09-08 - 5.3.0 - feat(web)

Add deployments API typings and web UI improvements: services & deployments management with CRUD and actions

- Add deployment request interfaces (`ts_interfaces/requests/deployment.ts`) to define typed API for create/read/update/delete/scale/restart operations.
- Extend web app state (`ts_web/appstate.ts`) to include typed services and deployments, and add actions for create/update/delete of services and deployments.
- Enhance web views (`ts_web/elements/*`): `CloudlyViewServices` and `CloudlyViewDeployments` now include richer display, styling, and UI actions (create, edit, deploy, restart, stop, delete).
- Fix subscription variable naming in several web components (subscription -> subscription) and improve table display functions to handle missing data safely.
- Add `.claude/settings.local.json` (tooling/permissions) used for local development/test tooling.

2025-09-07 - 5.2.0 - feat(settings)

Add runtime settings management, node & baremetal managers, and settings UI

- Introduce `CloudlySettingsManager` to store runtime settings in an `EasyStore` (MongoDB) with API handlers for get/update/clear/test.
- Add settings data/interface and typedrequest definitions (`ts_interfaces/data/settings.ts`, `ts_interfaces/requests/settings.ts`) and expose via interfaces index.
- Add web UI for managing provider credentials and connections (`ts_web/elements/cloudly-view-settings.ts`) and integrate the Settings view into the dashboard.
- Replace the previous `ServerManager` concept with `NodeManager` and `BaremetalManager`: new `ClusterNode` and `BareMetal` models and managers (auto-provisioning / Hetzner integration), plus `curlfresh` moved to node manager.
- Update Cluster data shape (servers -> nodes) and adjust related code paths (overview stats, cluster creation and provisioning flows).

- Use settingsManager for provider tokens (cloudflareToken, hetznerToken) instead of reading tokens directly from config/env; connector and manager init code updated accordingly.
- Add numerous implementations and API handlers to support baremetal/node lifecycle and control (getBaremetalServers, controlBaremetal, getNodeConfig, node provisioning helpers).
- Reorder Cloudly startup to initialize MongoDB and settings manager before managers that depend on settings; wire settingsManager into Cloudly class.
- Bump package dependency versions for @git.zone/tsdoc, @design.estate/dees-catalog and @push.rocks/taskbuffer in package.json.

2025-09-05 - 5.1.0 - feat(cluster)

Add cluster setupMode (manual|hetzner|aws|digitalocean) with conditional Hetzner auto-provisioning; UI and dashboard improvements; dependency upgrades

- Introduce optional setupMode on cluster configs and requests (ICluster.data.setupMode, createCluster request) to allow 'manual' | 'hetzner' | 'aws' | 'digitalocean'.
- ClusterManager: default setupMode to 'manual' when creating clusters and only trigger serverManager.ensureServerInfrastructure() for 'hetzner' clusters.
- ServerManager: skip provisioning for clusters not configured with setupMode 'hetzner' and log skipped clusters.
- Web UI: add a 'Setup Mode' dropdown when creating a cluster so users can choose auto-provisioning provider; ensure the add-cluster action passes setupMode.
- Web UI: dashboard enhancements — add icons to view tabs and replace cluster overview with a stats grid (including total clusters, total servers, images, services, deployments, secret groups/bundles, DNS, DBs, backups, mails, s3). The overview now computes total servers across clusters.
- Package dependency bumps (devDependencies and dependencies) to keep libs up-to-date (examples: @git.zone/tsbuild, @git.zone/tstest, @api.global/typedserver, @apiclient.xyz/docker, @design.estate/dees-catalog, @push.rocks/smartlog, @push.rocks/smartrequest, @push.rocks/taskbuffer, etc.).
- Add .claude/settings.local.json with local Claude permissions (editor/automation config).

2025-08-18 - 5.0.6 -

fix(connector.letsencrypt)

Improve Let's Encrypt integration and certificate handling; fix coreflow certificate response; add local assistant permissions config

- Replace ad-hoc setChallenge/removeChallenge hooks with a DNS-01 handler (smartacme.handlers.Dns01Handler) using Cloudflare to manage ACME DNS challenges.
- Add MongoDB-backed certificate manager (smartacme.certmanagers.MongoCertManager) and pass it to SmartAcme as certManager.
- Initialize SmartAcme with certManager and challengeHandlers instead of setChallenge/removeChallenge/mongoDescriptor options.
- Return certificate object directly from coreflow certificate request handler (avoid createSavableObject) to fix the getCertificateForDomain response payload.
- Add .claude/settings.local.json with local assistant/permissions entries to allow specific debugging/automation commands.
- Bump commitinfo versions to 5.0.6 and update changelog.

2025-08-18 - 5.0.6 - fix(connector.letsencrypt)

Improve Let's Encrypt integration and certificate handling; add local assistant permissions config

- Replace ad-hoc setChallenge/removeChallenge hooks with a DNS-01 handler using Cloudflare (smartacme.handlers.Dns01Handler) to manage ACME DNS challenges.
- Add MongoDB-backed certificate manager (smartacme.certmanagers.MongoCertManager) and pass it to SmartAcme as certManager.
- Update SmartAcme initialization to use certManager and challengeHandlers instead of setChallenge/removeChallenge/mongoDescriptor options.
- Return certificate object directly from coreflow certificate request handler (avoid createSavableObject), fixing the response payload for getCertificateForDomain.
- Add .claude/settings.local.json with local assistant/permissions entries to allow specific debugging/automation commands.

2025-08-18 - 5.0.5 - fix(coreflow)

Fix Coreflow identity lookup and response shape; improve API client tests and bump dependencies

- ts/manager.coreflow/coreflowmanager.ts: Use \$elemMatch to correctly query nested user.tokens when resolving identities and validate machine user types.
- ts/manager.coreflow/coreflowmanager.ts: Normalize getClusterConfig response to include services (was deploymentDirectives) and tidy handler signatures.
- test/test.apiclient.ts: Add detailed logging and improved error handling across preTask, client startup, identity retrieval, image creation and image upload to aid debugging and test observability.

- package.json: Update dependency versions (notable bumps): @types/node -> ^22.0.0, @push.rocks/smartacme -> ^8.0.0, @push.rocks/smartdata -> ^5.16.4, @push.rocks/smartexpect -> ^2.5.0, @push.rocks/smartpath -> ^6.0.0, @push.rocks/smartrequest -> ^4.2.2, plus other maintenance bumps.
- Add .claude/settings.local.json to provide local Claude permissions for developer tooling.

2025-04-25 - 5.0.4 - fix(platformservice/mta)

Update getEmailStatus response schema: make details property optional

- Changed details property from required with fixed message to optional with a flexible message structure in IReq_GetEmailStats response

2025-04-25 - 5.0.3 - fix(mta)

update email status response type in MTA platform service

- Changed the response 'status' field in IRequest_CheckEmailStatus from a literal 'unknown' to a generic string for improved flexibility

2025-04-25 - 5.0.2 - fix(platformservice/mta)

Refactor email status response in MTA service

- Updated IReq_CheckEmailStatus response: replaced union type ('ok' | 'not ok') with fixed status 'unknown' and added a details object with message 'Email not found'.

2025-04-25 - 5.0.1 - fix(mta)

Update email stats response interface in mta platform service to include totalEmailsSent, totalEmailsDelivered, totalEmailsBounced, averageDeliveryTimeMs, and lastUpdated timestamp.

- Modified IReq_GetEMailStats response in ts_interfaces/platformservice/mta.ts from an empty status object to a detailed email statistics structure.

2025-04-25 - 5.0.0 - BREAKING CHANGE(ts_interfaces/platformservice/mta)

Rename mta interfaces and upgrade dependency versions

- Upgraded devDependencies: @git.zone/tsbuild, tsbundle, tsdoc, ttest, tswatch, and @push.rocks/tapbundle to newer versions.
- Upgraded dependencies: @design.estate/dees-catalog, dees-domtools, dees-element, @push.rocks/smartdata, smartexpect, smartfile, smartpromise, smartrequest, smartrx, and tsclass (v4.2.0 to v9.0.0).
- Added new packageManager field in package.json and introduced pnpm-workspace.yaml for additional workspace configuration.
- Refactored mta API interfaces: renamed IRequest_SendEmail to IReq_SendEmail and IRequestRegisterRecipient to IReq_RegisterRecipient; added IReq_CheckEmailStatus and IReq_GetEMailStats.

2025-01-20 - 4.13.0 - feat(service)

Add support for service creation, update, and deletion.

- Implemented TypedHandlers for creating a new service.
- Added features to update existing service details.
- Enabled deletion of services by their unique ID.

2025-01-20 - 4.12.2 - fix(service)

Fix secret bundle and service management bugs

- Corrected the field name from 'includedImages' to 'imageClaims' in secret bundles.
- Implemented 'getFlatKeyValueObject' for secret bundles and modified related API interactions.

- Enhanced the Service class with methods for handling secret bundle data by resolving related groups and environments.

2025-01-02 - 4.12.1 - fix(deps)

Updated @git.zone/tspublish to version ^1.9.1

2025-01-02 - 4.12.0 - feat(cli)

Add CLI support and external registries view

- Adds CLI client functionality
- Introduces a new view for External Registries in the dashboard

2024-12-30 - 4.11.0 - feat(external-registry)

Introduce external registry management

- Added ExternalRegistryManager to handle external registry operations.
- Implemented ability to create, retrieve, and delete external registries.
- Enhanced Cloudly class to include ExternalRegistryManager.

2024-12-29 - 4.10.0 - feat(apiclient)

Added support for managing external registries in the API client.

- Introduced methods to get a registry by ID, get all registries, and create a new registry in the externalRegistry object.
- Updated external registry request interfaces to match new API client capabilities.

2024-12-29 - 4.9.0 - feat(apiclient)

Add external registry management capabilities to Cloudly API client.

- Introduce ExternalRegistry class with methods for getting, creating, and updating external registries.
- Expand requests module to handle external registry management, including creation and deletion.

2024-12-28 - 4.8.1 - fix(interfaces)

Fix image location schema in IImage interface

- Refactored the 'external' object within IImage data to a 'location' object.
- Added 'internal' boolean to 'location' to specify internal/external status.

2024-12-28 - 4.8.0 - feat(manager.registry)

Add external registry management

- Introduced ExternalRegistry class for handling external registry configurations.
- Updated IExternalRegistry interface to include registry details.
- Enhanced IImage interface to support linking with external registries.

2024-12-28 - 4.7.1 - fix(secretmanagement)

Refactor secret bundle actions and improve authorization handling

- Refactored secret bundle handling by renaming methods and reorganizing static and instance methods in SecretBundle class.
- Added getSecretBundleByAuthorization method to SecretBundle.

- Improved getFlatKeyValueObjectForEnvironment to accurately retrieve key-value pairs for specified environments.
- Removed deprecated IEnvBundle interface and related request handler for better clarity and code usage.
- Updated request interfaces related to secret bundles for consistent method naming and arguments.

2024-12-22 - 4.7.0 - feat(apiclient)

Add method to flatten secret bundles into key-value objects.

- SecretBundle: Implemented toFlatKeyValueObject method to flatten secret groups into key-value pairs.
- Removed stale SecretManager class from apiclient.

2024-12-22 - 4.6.0 - feat(cloudlyapiclient)

Extend CloudlyApiClient with cluster, secretbundle, and secretgroup methods

- Added methods to CloudlyApiClient for managing clusters: getClusterById, getClusters, createCluster.
- Added methods to CloudlyApiClient for managing secret bundles: getSecretBundleById, getSecretBundles, createSecretBundle.
- Added methods to CloudlyApiClient for managing secret groups: getSecretGroupById, getSecretGroups, createSecretGroup.

2024-12-22 - 4.5.5 - fix(apiclient)

Fixed image creation method in cloudlyApiClient

- Corrected method call from 'images.createImage' to 'image.createImage' to ensure proper image creation.
- Updated cluster retrieval methods and ensured proper API routes are being called.

2024-12-21 - 4.5.4 - fix(ts_web)

Fix action type and data fields in appstate for CRUD operations

- Correct request method in createSecretGroupAction to accurately reflect the purpose.
- Align the deleteSecretGroupAction and deleteSecretBundleAction request types with proper interfaces.
- Ensure data payload matches backend requirements for secret group and secret bundle operations.

2024-12-21 - 4.5.3 - fix(secret-management)

Refactor secret management to use distinct secret bundle and group APIs. Introduce API client classes for secret bundles and groups.

- Updated secret management logic to separate secret bundle and group APIs.
- Implemented new API client classes for managing secret bundles and groups.
- Fixed incorrect method usages for secret-related actions.

2024-12-20 - 4.5.2 - fix(apiclient)

Implemented IService interface in Service class and improved secret bundle documentation.

- Implemented plugins.servezoneInterfaces.data.IService interface in the Service class within ts_apiclient.
- Updated documentation comments for the type property in the ISecretBundle interface.

2024-12-17 - 4.5.1 - fix(core)

Updated dependencies in package.json to latest versions.

- Bumped @git.zone/tswatch to version ^2.0.37
- Bumped @types/node to version ^22.10.2
- Bumped @design.estate/dees-catalog to version ^1.3.2

- Bumped @push.rocks/smartfile to version ^11.0.23
- Bumped @tsclass/tsclass to version ^4.2.0

2024-12-14 - 4.5.0 - feat(services)

Add service management functionalities

- Integrated service-related API client methods including `getServices`, `getServiceById`, and `createService`.
- Updated the deployment data structure in the service manager.
- Enhanced service interface to incorporate additional fields for comprehensive data handling.
- Ensured secure token generation for Cloudly authentication processes.

2024-11-18 - 4.4.0 - feat(api-client)

Add static method `getImageById` for `Image` class in `api-client`

- Introduced a static method `getImageById` in the `Image` class.
- Updated `CloudlyApiClient` to include the `getImageById` method in the `images` interface.

2024-11-18 - 4.3.21 - fix(interfaces)

Remove deprecated deployment directive and update related interfaces

- Removed `IDeploymentDirective` from `data` and `requests`.
- Updated `IDeployment` to remove references to directives.
- Changed `IRequest_Any_Cloudly_GetClusterConfig` to return services instead of deployment directives.
- Removed `deploymentDirectives` from `IService` data structure.

2024-11-18 - 4.3.20 - fix(apiclient)

Ensure mandatory parameter in `CloudlyApiClient` constructor

- Made the 'optionsArg' parameter mandatory in the constructor of CloudlyApiClient class.

2024-11-18 - 4.3.19 - fix(docker)

Fix improper Docker push command preventing push to the correct registry.

- Corrected the docker push command in the '.gitea/workflows/docker_tags.yaml' file to push images to the 'code.foss.global' registry.

2024-11-17 - 4.3.18 - fix(docker_tags)

Updated Docker configuration to include NPM tokens.

- Restored NPMCI_TOKEN_NPM and NPMCI_TOKEN_NPM2 environment variables in docker_tags.yaml for authentication purposes.

2024-11-17 - 4.3.17 - fix(Dockerfile)

Corrected docker base image tag in Dockerfile for alpine compatibility.

- Updated Dockerfile to use the correct base image tag for Alpine.
- Resolved any potential build issues related to incorrect image tag usage.

2024-11-17 - 4.3.16 - fix(infrastructure)

Correct Docker image path in Dockerfile for improved build consistency.

- Dockerfile: Updated base image paths from 'code.foss.global/hosttoday/ht-docker-node' to 'code.foss.global/host.today/ht-docker-node'.

2024-11-17 - 4.3.15 - fix(project setup)

fixed incorrect configuration in npmextra.json

- Removed unnecessary 'dockerBuildargEnvMap' entry for NPMCI_TOKEN_NPM2.
- Corrected the githost and gitscope in gitzone module configuration.
- Updated the license field to reflect the correct license.

2024-11-16 - 4.3.14 - fix(docker tags)

Comment out unused secret variables in docker_tags.yaml

- Modified docker_tags.yaml to comment out unused secret variables related to NPM and GitHub tokens.

2024-11-16 - 4.3.13 - fix(package)

Updated package dependencies

- Updated @design.estate/dees-catalog version to 1.3.1

2024-11-06 - 4.3.12 - fix(workflow)

Fix Docker image path in GitHub action workflow

- Corrected the path of the Docker image used in the GitHub action workflow from 'code.foss.global/hosttoday/ht-docker-dbase:npmci' to 'code.foss.global/host.today/ht-docker-dbase:npmci'.

2024-11-06 - 4.3.11 - fix(overall)

Refactor and improve code consistency across all modules

- Updated cloud configuration management for better reliability.
- Enhanced security measures in the authentication and authorization processes.
- Streamlined deployment logic in cluster management.
- Refactored code to improve maintainability and readability.

2024-11-06 - 4.3.10 - fix(dependencies)

Updated dependencies and fixed Docker Alpine image retrieval issue in tests

- Updated @push.rocks/tapbundle to version ^5.5.0 in devDependencies.
- Updated @push.rocks/smartrequest to version ^2.0.23 in dependencies.
- Ensured the Docker Alpine image is retrieved as a local tarball in cloudlyfactory.ts test helper.

2024-11-06 - 4.3.9 - fix(test and dependencies)

Corrected cloudlyUrl in test.apiclient and updated tapbundle dependency.

- Fixed cloudlyUrl assignment in the test.apiclient to use the correct helper method.
- Updated tapbundle dependency version from ^5.4.3 to ^5.4.4.

2024-11-06 - 4.3.8 - fix(api client)

Fixed localhost URL issue in test.client.ts

- Changed the cloudlyUrl in test.client.ts from 'localhost' to '127.0.0.1' to ensure consistency in network requests.

2024-11-06 - 4.3.7 - fix(tests)

Refactored test setup for consistency and isolated config initialization.

- test/helpers/cloudlyfactory.ts: Test configuration setup was refactored to ensure consistent initialization of cloudly configuration across tests.
- test/test.apiclient.ts: Updated cloudlyApiClient test setup to use testCloudlyConfig for dynamic port allocation.

2024-11-06 - 4.3.6 - fix(test)

Enhance test helpers with dynamic Hetzner token retrieval.

- Updated test/helpers/cloudlyfactory.ts to retrieve Hetzner token from environment variables.
- Expanded docker_tags workflow to securely handle and pass new environment secrets.

2024-11-06 - 4.3.5 - fix(helper)

Add missing sslMode configuration to Cloudly config.

- Added the sslMode key with a value of 'none' to the Cloudly configuration object in test/helpers/cloudlyfactory.ts.

2024-11-06 - 4.3.4 - fix(testing)

Fixed Cloudly testing setup and dependencies

- Updated devDependency @push.rocks/tapbundle version from ^5.3.0 to ^5.4.3 in package.json.
- Updated devDependency @push.rocks/npmextra version from ^5.1.1 to ^5.1.2 in package.json.
- Improved the Cloudly test suite setup to ensure proper initialization of MongoDB and S3 services.
- Ensured asynchronous stopping and cleanup of MongoDB and S3 services post-test execution.

2024-11-05 - 4.3.3 - fix(core)

Fix configuration initialization by accepting a config argument

- Configuration initialization now accepts an optional config argument
- Updated test cloudly factory to use default public URL and port
- Updated dependencies versions

2024-11-05 - 4.3.2 - fix(npmextra)

Updated npm registry URL in npmextra.json

2024-11-05 - 4.3.1 - fix(package)

Update dependency version for @git.zone/tspublish

- Bump @git.zone/tspublish from version ^1.7.6 to ^1.7.7 in package.json

2024-11-05 - 4.3.0 - feat(dependencies)

Upgrade dependencies and include publish orders

- Upgraded @git.zone/tsbuild to version ^2.2.0
- Upgraded @git.zone/tspublish to version ^1.7.6
- Upgraded @types/node to version ^22.8.7
- Added publish order to ts_apiclient/tspublish.json and ts_interfaces/tspublish.json

2024-11-04 - 4.2.1 - fix(config)

Fix Docker image URL in Gitea workflow.

- Corrected the IMAGE URL from 'hosttoday' to 'host.today'.

2024-11-04 - 4.2.0 - feat(cloudron)

Add Dockerfile for Cloudron deployment

- Introduced a new Dockerfile for Cloudron deployment.
- The Dockerfile uses the latest version of cloudly as a base image.

2024-10-28 - 4.1.3 - fix(dependency)

Updated dependency @git.zone/tspublish to version ^1.6.0

- Bumped @git.zone/tspublish version from ^1.5.5 to ^1.6.0 in package.json

2024-10-28 - 4.1.2 - fix(core)

Corrected description and devDependencies

- Updated package.json description to accurately reflect features.
- Added @git.zone/tsdoc to devDependencies.
- Corrected version discrepancy in @types/node devDependency.
- Standardized description across multiple files including npmextra.json.

2024-10-28 - 4.1.1 - fix(core)

Fixed syntax issues in commitinfo data and package.json file.

- Added a missing newline at the end of the package.json file.
- Corrected a trailing comma and added proper syntax in the commitinfo data.

2024-10-28 - 4.1.0 - feat(core)

Enhance core functionality for cloud management and orchestration

- Improved initialization and management of cloud environments with Docker Swarmkit.
- Added capability to manage DNS records via Cloudflare.
- Introduced integration support for DigitalOcean resources.

2024-10-28 - 4.0.1 - fix(package_manager)

Update @git.zone/tspublish dependency version

- Bump @git.zone/tspublish version from 1.5.4 to 1.5.5.

2024-10-28 - 4.0.0 - BREAKING CHANGE(core)

Significant overhaul with potential breaking changes, update to version 3.0.0

- Updated project version from 1.2.5 to 3.0.0 in package.json

2024-10-28 - 1.2.5 - fix(build)

Updated devDependencies for tspublish and removed buildDocs script

- devDependencies updated: @git.zone/tspublish to version ^1.5.4
- Removed: buildDocs script from the scripts section

2024-10-27 - 1.2.4 - fix(ci)

Fix Docker images and npm registry URL in CI workflows

- Updated Docker image registry URL from 'registry.gitlab.com' to 'code.foss.global'.
- Fixed npmci package installation path from '@shipzone/npmci' to '@ship.zone/npmci'.

2024-10-23 - 1.2.3 - fix(cli)

Set up CLI client definition and registry configuration

- Added a console log message for CLI identification.
- Defined package name for CLI client.
- Configured npm registries for package distribution.

2024-10-23 - 1.2.2 - fix(docs)

Updated documentation with clearer usage instructions and examples.

- Refined setup guides in README.
- Added more detailed examples in code snippets.
- Clarified cloud service integration instructions.

2024-10-23 - 1.2.1 - fix(core)

Fixed startup issue for the Cloudly instance

2024-10-21 - 1.2.0 - feat(cli)

Add tspublish.json for CLI client and interfaces

- Added registration details for publishing CLI client (@serve.zone/cli)
- Specified npm and custom registries in the tspublish.json for better publish control
- Updated dependency version for @git.zone/tspublish in package.json

2024-10-21 - 1.1.9 - fix(build)

Update Node types and other dependencies, add tspublish.json for api client

- Updated Node types devDependency to version ^22.7.7 from ^22.7.5.
- Updated smartbucket dependency to version ^3.0.23 from ^3.0.22.
- Added tspublish configuration file to the api client.

- Fixed the npm publish script in package.json.

2024-10-16 - 1.1.8 - fix(big fix upgrade)

fix: update dependency versions and address type errors

- Updated all listed dependencies in the package.json to their specified ranges.
- Fixed type mismatches and added missing imports in various TypeScript files.
- Refined existing tests and added a new helper to manage Docker image streams.

2024-08-25 - 1.1.7 - fix(deps)

Update dependencies to latest versions

- Updated @git.zone/tsbuild from ^2.1.80 to ^2.1.84
- Updated @push.rocks/tapbundle from ^5.0.23 to ^5.0.24
- Updated @types/node from ^20.14.6 to ^22.5.0
- Updated @apiclient.xyz/docker from ^1.2.2 to ^1.2.3
- Updated @design.estate/dees-catalog from ^1.0.289 to ^1.1.6
- Updated @design.estate/dees-element from ^2.0.34 to ^2.0.36
- Updated @git.zone/tsrun from ^1.2.37 to ^1.2.49
- Updated @push.rocks/smartbucket from ^3.0.20 to ^3.0.22
- Updated @push.rocks/smartpromise from ^4.0.3 to ^4.0.4
- Updated @serve.zone/interfaces from ^1.0.74 to ^1.0.78
- Updated @tsclass/tsclass from ^4.0.60 to ^4.1.2

2024-06-20 - 1.1.6 - Updates

Routine updates and fixes.

- (fix) core: update

2024-06-13 - 1.1.4 - Service Management Preparation

Incorporated updates and service management preparations.

- (fix) core: update
- (feat) prepare service management

2024-06-05 - 1.1.3 - CI Integration Improvement

Structural improvements and better CI integration preparation.

- (fix) structure: improve structure, prepare better CI integration

2024-06-02 - 1.1.2 - Image Manager Update

Prepared proper storage and retrieval of container images.

- (fix) imagemanager: prepare proper storage and retrieval of container images

2024-06-01 - 1.1.0 - Image Registry Work

Initiated work on image registry.

- (fix) image registry: start work on image registry

2024-05-30 - 1.0.216 - Enhanced Smartguards

Enhanced smartguards to verify action authorization.

- (feat) guards: use better smartguards to verify action authorization

2024-05-28 - 1.0.215 - Unified Package Update

Updated package unification for cloudly + API + CLI.

- (fix) switch to unified package for cloudly + API + CLI: update

2024-05-05 - 1.0.214 - Core Updates

Routine core updates.

- (fix) core: update

2024-04-20 - 1.0.213 - Core Update

Routine core updates.

- (fix) core: update