Unified Configuration Schema Reference

Unified Configuration Schema Reference

Overview

This document provides the complete configuration schema for the unified cluster-snek platform, combining the simplicity of cluster-snek with the comprehensive service stack from homelab-ai-rnd-stack.

Core Configuration Structure

```
# Project identification
project_name: string (required)
environment: string (default: "production")

# Deployment configuration
deployment_mode: enum (default: "internet")
  - internet
  - airgapped-vc
  - airgapped-local
  - airgapped-network
  - airgapped-archive

deployment_target: enum (default: "vms")
  - vms
  - direct
  - hybrid

# Cluster definitions
clusters: list[ClusterConfig] (required)

# Service configuration
services: list[string] (optional)
service_overrides: dict (optional)
service_profile: string (optional)

# Security configuration
security_profile: enum (optional)
  - minimal
  - standard
  - hardened
  - zero-trust

# Global settings
base_domain: string (default: "example.com")
metallb_ip_range: string (default: "192.168.1.200-192.168.1.250")
github_organization: string (optional)
auto_create_repositories: boolean (default: true)
```

Cluster Configuration

```
clusters:
  - name: string (required)
    domain: string (required)
```

```
    size: enum (default: "small")
      - minimal   # 1 node, 2 CPU, 4GB RAM
      - small     # 2-3 nodes, 4 CPU, 8GB RAM
      - medium    # 3-5 nodes, 8 CPU, 16GB RAM
      - large     # 5+ nodes, 16+ CPU, 32+ GB RAM

    # Features
    gpu_enabled: boolean (default: false)
    vector_store: enum (default: "disabled")
      - disabled
      - weaviate
      - qdrant
      - chroma
      - chroma-memory

    cerbos_enabled: boolean (default: false)

    # Workloads
    specialized_workloads: list[string] (optional)
      - development
      - testing
      - machine-learning
      - ai-inference
      - security
      - monitoring

    # Services (cluster-specific)
    services: list[string] (optional)

    # Custom values
    custom_values: dict (optional)
```

Service Configuration

Available Services

```
services:
  # Core Infrastructure
  - keycloak         # SSO and identity management
  - gitlab           # Source control and CI/CD
  - cert-manager     # Certificate automation

  # AI/ML Platform
  - jupyterhub       # Notebook environment
  - weaviate         # Vector database
  - qdrant           # Alternative vector DB
  - mlflow           # ML experiment tracking

  # Monitoring & Observability
  - monitoring       # Prometheus/Grafana/Loki stack
  - prometheus       # Metrics collection only
  - grafana          # Visualization only
  - loki             # Log aggregation only

  # Security & Compliance
  - cerbos           # Authorization engine
  - vault            # Secret management
  - falco            # Runtime security
```

```
  # Backup & Recovery
  - velero           # Backup solution

  # Networking
  - istio            # Service mesh
  - linkerd          # Alternative service mesh
```

Service Overrides

```yaml
service_overrides:
  gitlab:
    version: "16.7.0"
    replicas: 3
    storage:
      repos: "200Gi"
    custom_values:
      gitlab:
        webservice:
          minReplicas: 3

  monitoring:
    components:
      - prometheus
      - grafana
      - alertmanager
    prometheus:
      retention: "90d"
      storage: "200Gi"
```

Security Profiles

Minimal

Basic security for development environments.

```yaml
security_profile: "minimal"
# Includes:
# - Basic RBAC
# - Default network policies
# - Standard pod security
```

Standard

Recommended for most deployments.

```yaml
security_profile: "standard"
# Includes:
# - Enhanced RBAC
# - Restricted network policies
# - Pod security standards
# - Secret encryption at rest
```

Hardened

Enhanced security for sensitive workloads.

```yaml
security_profile: "hardened"
```

```
# Includes:
# - Strict RBAC
# - Default-deny network policies
# - Enforced pod security
# - mTLS everywhere
# - Audit logging
```

Zero-Trust

Maximum security with zero trust principles.

```
security_profile: "zero-trust"
# Includes:
# - All hardened features
# - Service mesh with mTLS
# - Cerbos authorization
# - Network segmentation
# - Continuous compliance monitoring
```

Source Configuration (Airgapped)

```
source:
  mode: enum (required for airgapped)
    - airgapped-vc
    - airgapped-local
    - airgapped-network
    - airgapped-archive

  # For airgapped-vc
  url: string
  username: string
  token: string
  branch: string (default: "main")

  # For airgapped-local
  path: string

  # For airgapped-network
  url: string
  username: string (optional)
  password: string (optional)

  # For airgapped-archive
  path: string
  verification_enabled: boolean (default: true)
  checksum: string (optional)
```

Global Overrides

```
global_overrides:
  # Domain settings
  domain: string
  timezone: string (default: "UTC")

  # Storage
  storageClass: string (default: "openebs-hostpath")

```

```
  # Monitoring
  monitoring:
    enabled: boolean (default: true)
    retention: string (default: "30d")
    alerting:
      enabled: boolean
      receivers:
        - type: email
          config: {...}
        - type: slack
          config: {...}

  # Security
  security:
    networkPolicies: boolean (default: true)
    podSecurityStandards: boolean (default: true)
    defaultDenyAll: boolean (default: false)
    encryptionAtRest: boolean (default: true)
    encryptionInTransit: boolean (default: true)

  # Compliance
  compliance:
    frameworks:
      - cis-kubernetes
      - nist-800-53
      - iso-27001
    audit_logging: boolean (default: true)
    policy_enforcement: string (default: "warn")

  # Image registry
  imageRegistry:
    internal: string (optional)
    pullPolicy: string (default: "IfNotPresent")
    imagePullSecrets:
      - name: string
```

Complete Example

Production AI Platform

```
project_name: "vectorweight-ai-platform"
environment: "production"
deployment_mode: "internet"
deployment_target: "vms"

clusters:
  - name: "ai-prod"
    domain: "ai.vectorweight.com"
    size: "large"
    gpu_enabled: true
    vector_store: "weaviate"
    cerbos_enabled: true
    specialized_workloads:
      - machine-learning
      - ai-inference
      - model-training
    services:
      - keycloak
```

```
      - gitlab
      - jupyterhub
      - weaviate
      - monitoring
      - velero
      - cerbos

  - name: "ai-dev"
    domain: "ai-dev.vectorweight.com"
    size: "small"
    gpu_enabled: false
    vector_store: "chroma-memory"
    services:
      - jupyterhub
      - monitoring

security_profile: "zero-trust"

service_overrides:
  weaviate:
    replicas: 5
    modules:
      - text2vec-transformers
      - generative-openai
      - qna-transformers
  gitlab:
    runners:
      count: 10
      gpu_enabled: true

github_organization: "vectorweight"
auto_create_repositories: true
base_domain: "vectorweight.com"
metallb_ip_range: "10.0.100.200-10.0.100.250"

global_overrides:
  monitoring:
    retention: "90d"
    alerting:
      enabled: true
      receivers:
        - type: slack
          config:
            webhook: "${SLACK_WEBHOOK}"
  security:
    defaultDenyAll: true
  compliance:
    frameworks:
      - cis-kubernetes
      - iso-27001
```

Environment Variables

The following environment variables can override configuration:

```
# GitHub integration
GITHUB_TOKEN
GITHUB_ORGANIZATION
```

```
4
5  # Domain configuration
6  BASE_DOMAIN
7  METALLB_IP_RANGE
8
9  # Security
10 ENABLE_ZERO_TRUST
11 SECURITY_PROFILE
12
13 # Services
14 ENABLE_MONITORING
15 ENABLE_BACKUP
16
17 # Deployment
18 DEPLOYMENT_MODE
19 DEPLOYMENT_TARGET
```

Validation Rules

1. **Required Fields**:
   - project_name
   - At least one cluster
   - Cluster name and domain
2. **Naming Conventions**:
   - Cluster names: lowercase alphanumeric with hyphens
   - Domains: valid FQDN format
   - Project names: alphanumeric with hyphens
3. **Resource Constraints**:
   - Minimal clusters: max 2 services
   - Small clusters: max 5 services
   - GPU requires medium or large size
4. **Service Dependencies**:
   - JupyterHub requires Keycloak
   - Cerbos audit requires PostgreSQL
   - Velero requires object storage

Migration from Legacy Formats

From cluster-snek v1

```
1 vectorweight migrate --from cluster-snek-v1 --config old.yaml
```

From homelab-ai-rnd-stack

```
1 vectorweight migrate --from homelab --source /path/to/homelab
```

Manual migration

Use the configuration converter tool:

```
1 vectorweight config convert --input old-format.yaml --output new-format.yaml
```

Best Practices

1. **Start Simple**: Begin with minimal configuration and add features
2. **Use Profiles**: Leverage service and security profiles
3. **Environment Separation**: Use different configs per environment
4. **Version Control**: Store configurations in Git
5. **Validate Often**: Run `vectorweight validate` before deployment
6. **Document Overrides**: Comment custom configurations