

Hydra Infrastructure

Complete Reference Manual

RKE2 Kubernetes Cluster
Student Container Platform
GPU Computing Infrastructure

SUNY New Paltz — Computer Science Department

Infrastructure Team

February 9, 2026

Contents

| | | |
|-----------|--|-----------|
| I | System Overview | 6 |
| 1 | Introduction | 6 |
| 1.1 | Key Features | 6 |
| 1.2 | Access URLs | 6 |
| 2 | Cluster Architecture | 7 |
| 2.1 | Node Inventory | 7 |
| 2.2 | Architecture Diagram | 7 |
| 2.3 | Network Architecture | 7 |
| 3 | Storage | 8 |
| 3.1 | Hydra Storage Layout | 8 |
| 3.2 | Kubernetes Storage Classes | 8 |
| 3.3 | NFS Configuration | 8 |
| II | Kubernetes Services | 9 |
| 4 | Namespace Layout | 9 |
| 5 | Core Services (hydra-system) | 9 |
| 5.1 | Traefik (Reverse Proxy) | 9 |
| 5.2 | Hydra Auth (SAML Gateway) | 9 |
| 5.3 | CS Lab Website | 10 |
| 5.4 | IngressRoute Summary | 10 |
| 6 | Infrastructure Services (hydra-infra) | 10 |
| 6.1 | Ollama (LLM Inference) | 10 |
| 6.2 | OpenWebUI | 10 |
| 6.3 | n8n (Workflow Automation) | 11 |

| | | |
|------------|-------------------------------------|-----------|
| 6.4 | Ray Cluster (Distributed Computing) | 11 |
| 6.5 | Other Services | 11 |
| 7 | GPU Infrastructure | 11 |
| 7.1 | NVIDIA GPU Operator | 11 |
| 7.2 | GPU Allocation | 12 |
| III | Authentication System | 13 |
| 8 | SAML 2.0 SSO Flow | 13 |
| 9 | Session and JWT Management | 13 |
| 10 | Cross-Service Authentication | 13 |
| 10.1 | OpenWebUI Account Provisioning | 13 |
| 10.2 | n8n Account Provisioning | 13 |
| 10.3 | CS Lab JWT Verification | 14 |
| IV | Student Containers | 15 |
| 11 | Container Features | 15 |
| 11.1 | Service Management | 15 |
| 12 | Container Presets | 15 |
| 13 | Resource Presets | 16 |
| 14 | Pod Timing and Lifecycle | 16 |
| 14.1 | Duration Tiers | 16 |
| 14.2 | Resource Limits | 16 |
| 14.3 | Auto-Approval Thresholds | 16 |
| 14.4 | K8s Resource Quotas per Preset | 17 |
| 14.5 | Max Concurrent Pods per Node | 17 |
| 14.6 | Pod Lifecycle Settings | 17 |
| 14.7 | JWT Session Duration | 17 |
| 15 | SSH Access via SSHPiper | 18 |
| 16 | Container Labels and Routing | 18 |
| V | Networking | 19 |
| 17 | Firewall Configuration (UFW) | 19 |
| 17.1 | Hydra (Control Plane) | 19 |
| 17.2 | Chimera (GPU Worker) | 19 |
| 17.3 | Cerberus (GPU Worker) | 19 |
| 18 | Router Port Forwarding | 19 |
| 19 | DNS | 19 |

| | | |
|-------------|--|-----------|
| VI | Deployment and Operations | 21 |
| 20 | Ansible Playbooks | 21 |
| 20.1 | Playbook Execution Order | 21 |
| 20.2 | What 05-deploy-hydra.yml Deploys | 21 |
| 21 | CS Lab Website Deployment | 21 |
| 22 | Image Management | 22 |
| 23 | Backup System | 22 |
| 23.1 | Daily Cluster Backups | 22 |
| 23.2 | etcd Snapshots | 22 |
| 23.3 | Backup Exclusions | 22 |
| 24 | Automation and Scheduled Tasks | 22 |
| 24.1 | System Cron Jobs | 23 |
| 24.2 | Application Background Services | 23 |
| 24.3 | Security Monitor Thresholds | 23 |
| 24.4 | Resource Expiry Behavior | 23 |
| 24.5 | Dynamic Route Management | 24 |
| 24.6 | Environment Variables for Automation | 24 |
| 25 | Common Operations | 24 |
| 25.1 | Kubectl Shortcuts | 24 |
| 25.2 | Service Management | 25 |
| VII | Web Services and Route Map | 26 |
| 26 | Complete Site Inventory | 26 |
| 27 | Traefik Route Priority Table | 26 |
| 28 | How to Update Each Service | 27 |
| 28.1 | Hydra Auth (Dashboard) | 27 |
| 28.2 | CS Lab Website | 27 |
| 28.3 | Hackathons | 27 |
| 28.4 | FLAPJS | 27 |
| 28.5 | Git Learning | 28 |
| 28.6 | Java Executor | 28 |
| 28.7 | Student Container Image | 28 |
| 28.8 | OpenWebUI and n8n | 28 |
| 29 | K8s Manifests Location | 28 |
| 30 | Namespace Layout | 29 |
| VIII | OpenWebUI API Integration | 30 |
| 31 | Getting Started | 30 |
| 32 | API Configuration | 30 |

| | |
|--|---------------|
| 33 cURL Example | 30 |
| 34 Python Example | 30 |
| 35 JavaScript Example | 30 |
| IX Security | 32 |
| 36 Security Architecture Layers | 32 |
| 37 Known Vulnerabilities | 32 |
| 38 Security Best Practices for Students | 32 |
| X RDMA and GPUDirect | 33 |
| 39 Overview | 33 |
| 40 Installation Order (Critical) | 33 |
| 41 SoftRoCE Setup | 33 |
| 42 GPUDirect RDMA Verification | 33 |
| XI Troubleshooting | 35 |
| 43 Authentication Issues | 35 |
| 44 Container Issues | 35 |
| 45 GPU Issues | 35 |
| 46 Networking Issues | 36 |
| 47 Traefik Deployment Issues | 36 |
| 48 CS Lab Website Catch-All Route | 36 |
| XII Repository Structure | 37 |
| 49 hydra-saml-auth | 37 |
| 50 Other Repositories | 38 |
| XIII Environment Configuration | 39 |
| 51 Required Variables (hydra-saml-auth) | 39 |
| 52 Ansible Inventory Variables | 39 |

| | |
|---|-----------|
| Appendices | 40 |
| A Cleanup History (February 2026) | 40 |
| B Migration History | 40 |
| C February 9, 2026 — Infrastructure Overhaul | 40 |
| D February 9, 2026 — Jenkins Service + Jupyter Gating + Repo Cleanup | 41 |
| E February 9, 2026 — Route Fixes + Pod Restart + Documentation | 41 |
| F References | 42 |

Part I

System Overview

1 Introduction

Hydra is a containerized development platform providing persistent development environments for Computer Science students and faculty at SUNY New Paltz. The system runs on a 3-node RKE2 Kubernetes cluster with GPU acceleration for AI/ML workloads.

1.1 Key Features

- **SSO Authentication:** Azure AD SAML 2.0 with automatic user provisioning
- **Persistent Containers:** Per-student development environments with SSH, VS Code, and Jupyter
- **GPU Computing:** 5 GPUs across 2 nodes (3x RTX 3090 + 2x RTX 5090)
- **AI Chat:** OpenWebUI + Ollama LLM inference (gpt.hydra.newpaltz.edu)
- **Ray Cluster:** Distributed computing framework for ML training
- **Dynamic Routing:** Traefik reverse proxy with ForwardAuth
- **Workflow Automation:** n8n with integrated user management
- **21 TB Storage:** RAID-10 ZFS array with NFS exports

1.2 Access URLs

| Service | URL | Description |
|---------------|---|---------------------|
| Dashboard | https://hydra.newpaltz.edu/dashboard | Main user interface |
| OpenWebUI | https://gpt.hydra.newpaltz.edu/ | AI chat (Ollama) |
| CS Lab Site | https://hydra.newpaltz.edu/ | Department homepage |
| Student Forms | https://hydra.newpaltz.edu/student-forms | Form hub |
| Hackathons | https://hydra.newpaltz.edu/hackathons | Hackathon voting |
| VS Code | https://hydra.newpaltz.edu/students/{user}/vscode | Browser IDE |
| Jupyter | https://hydra.newpaltz.edu/students/{user}/jupyter | Notebooks |
| n8n | https://n8n.hydra.newpaltz.edu/ | Workflow automation |
| Servers | https://hydra.newpaltz.edu/servers | Cluster status |

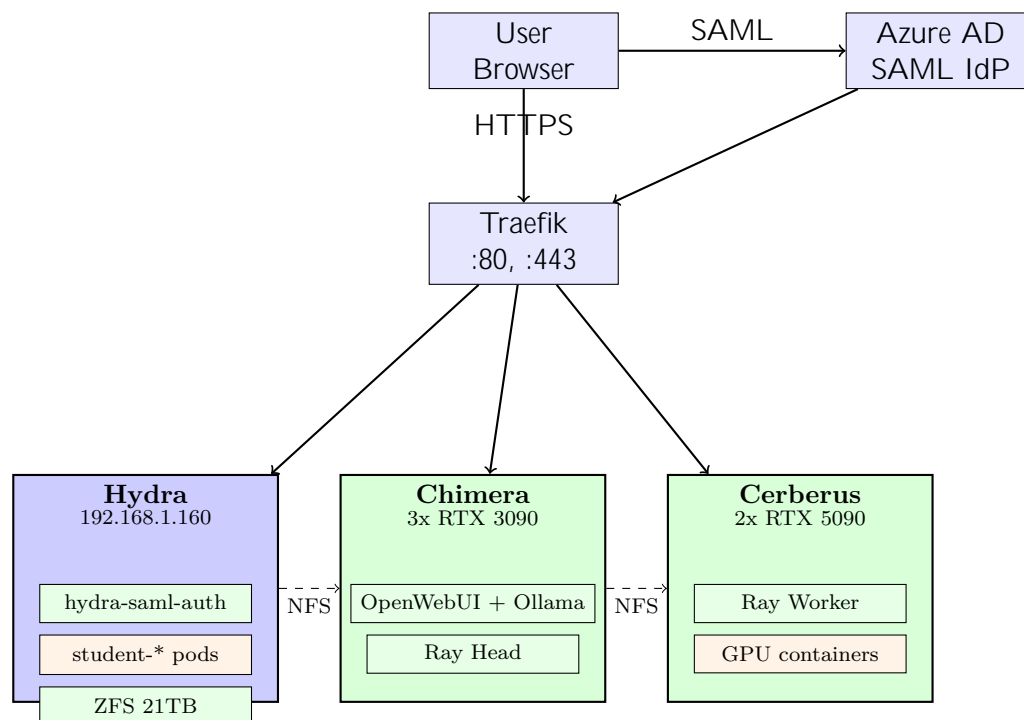
2 Cluster Architecture

2.1 Node Inventory

| Node | IP | Role | OS | Hardware |
|----------|---------------|---------------------|----------------|--|
| Hydra | 192.168.1.160 | Control plane, etcd | Ubuntu 22.04.5 | 64 cores, 256 GB RAM, 21 TB RAID-10 |
| Chimera | 192.168.1.150 | GPU inference | Ubuntu 24.04.2 | 48 cores, 256 GB RAM, 3x RTX 3090 (72 GB VRAM) |
| Cerberus | 192.168.1.242 | GPU training | Ubuntu 24.04.3 | 48 cores, 64 GB RAM, 2x RTX 5090 (64 GB VRAM) |

Table 1: All nodes run RKE2 v1.28.4+rke2r1 with containerd 1.7.7.

2.2 Architecture Diagram



2.3 Network Architecture

- All nodes on 192.168.1.0/24 LAN (gateway 192.168.1.1)
- Direct ethernet bridge between Chimera and Cerberus (reserved for RDMA/RoCE)
- WireGuard VPN: Chimera `wg0` = 10.8.0.2, Cerberus `wg0` = 10.8.0.3
- Flannel VXLAN (port 8472/udp) for K8s pod networking, restricted to LAN
- UFW firewall on all nodes — workers expose only SSH publicly

3 Storage

3.1 Hydra Storage Layout

| Device | Mount | Size | Purpose |
|----------------------------|-----------|--------|---------------------------|
| /dev/mapper/ubuntu-vg-* | / | 1 TB | OS, applications |
| /dev/md0 (RAID-10, 6 SSDs) | /data | 21 TB | Student volumes, K8s PVCs |
| /dev/sdh4 | /mnt/sdh4 | 1.1 TB | Daily backups |

```
# RAID-10 details
/dev/md0: 6 active devices (sda-sdf), Chunk 512K, Layout near=2
State: clean, ext4, 4096-byte blocks
```

3.2 Kubernetes Storage Classes

| Name | Provisioner | Usage |
|-------------|-----------------------|---------------------------|
| hydra-local | rancher.io/local-path | Student PVCs (default) |
| hydra-nfs | nfs.csi.k8s.io | Cross-node shared storage |

3.3 NFS Configuration

Hydra exports /data/containers to the cluster LAN:

```
# /etc/exports on Hydra
/data/containers 192.168.1.0/24(rw, sync, no_root_squash)
```

CSI-NFS runs as a DaemonSet on all 3 nodes for dynamic PV provisioning.

Part II

Kubernetes Services

4 Namespace Layout

| Namespace | Contents |
|--------------------|---|
| hydra-system | Core platform: traefik, hydra-auth, cs-lab-backend, cs-lab-db |
| hydra-infra | Infrastructure services: ollama, open-webui, n8n, hackathons, java-executor, git-learning, sshpipec, ray-head, ray-worker |
| hydra-students | Student container pods (25+ active) |
| gpu-operator | NVIDIA GPU operator, device plugin, DCGM exporter |
| kube-system | RKE2 system: etcd, coredns, canal, metrics-server, CSI-NFS |
| local-path-storage | Local-path provisioner |

5 Core Services (hydra-system)

5.1 Traefik (Reverse Proxy)

Traefik v2.11 serves as the cluster ingress controller. It runs on Hydra with `hostPort` binding on ports 80, 443, and 6969. The deployment uses `strategy: Recreate` to avoid `hostPort` conflicts during rolling updates.

| Port | Name | Purpose |
|------|------------|----------------------------|
| 80 | web | HTTP (redirects to HTTPS) |
| 443 | websecure | HTTPS with Let's Encrypt |
| 6969 | hydra-auth | Direct auth service access |

Manifests: `k8s/components/traefik/`

5.2 Hydra Auth (SAML Gateway)

The main authentication and container management service. Handles:

- SAML 2.0 SSO via Azure AD
- JWT cookie issuance and JWKS endpoint
- Student container lifecycle (create, start, stop, delete)
- Dashboard UI, admin panel
- OpenWebUI and n8n account provisioning
- WebSocket terminal bridge

Manifests: `k8s/components/hydra-auth/`

5.3 CS Lab Website

React frontend + Express backend + SQLite database (single pod). Serves the department homepage at `hydra.newpaltz.edu`. MariaDB was removed on Feb 9, 2026 — the codebase uses `better-sqlite3` exclusively.

| Component | Port | Image |
|-----------|------|--|
| cs-lab | 5001 | newpaltz-cs-lab-website-backend:latest |

Database: SQLite at `/app/server/data/cs1ab.db`. 15 tables including Admins, Events, Faculty, Courses, StudentHighlightBlog, TechBlog, etc. Persisted via PVC `cs-lab-data`.

Manifests: `k8s/components/cs-lab/`

5.4 IngressRoute Summary

| Name | Namespace | Match | Backend |
|----------------|--------------|------------------------------|------------------------|
| hydra-main | hydra-system | hydra.newpaltz.edu catch-all | hydra-auth:6969 |
| cs-lab-website | hydra-system | /api/ prefix | cs-lab-backend:5001 |
| hydra-default | hydra-system | HTTP redirect | HTTPS redirect |
| hackathons | hydra-infra | /hackathons/ prefix | hackathons:45821 |
| java-executor | hydra-infra | /java/ prefix | java-executor:55392 |
| git-learning | hydra-infra | /git/ prefix | git-learning:8080 |
| n8n | hydra-infra | n8n.hydra.newpaltz.edu | n8n:5678 |
| openwebui | hydra-infra | gpt.hydra.newpaltz.edu | openwebui-chimera:3000 |

6 Infrastructure Services (hydra-infra)

6.1 Ollama (LLM Inference)

Runs on Chimera with all 3 RTX 3090 GPUs. Serves LLM models (gemma3:12b, etc.) via the Ollama API on port 11434.

Manifests: `k8s/components/ollama/`

Ollama requests all 3 GPUs on Chimera. Other GPU workloads on Chimera (like Ray head) must **not** request GPU resources, or they will conflict.

6.2 OpenWebUI

AI chat frontend at `gpt.hydra.newpaltz.edu`. Connects to Ollama for inference. Includes a **middleman sidecar** container for user account management.

Middleman API (port 7070):

- POST `/openwebui/api/check-user` — Check if user exists
 - POST `/openwebui/api/create-account` — Create new user
 - POST `/openwebui/api/change-password` — Update password
- Authentication via `x-api-key` header with timing-safe comparison.

Source: `k8s/components/openwebui/middleman/index.js`

Manifests: `k8s/components/openwebui/`

6.3 n8n (Workflow Automation)

Workflow automation platform at `n8n.hydra.newpaltz.edu`. Uses PostgreSQL for data storage.

Components:

- n8n application (port 5678)
- PostgreSQL 16 (StatefulSet with PVC)
- n8n User Manager API (port 3000)

n8n User Manager API:

- GET `/health` — Health check (no auth)
 - GET `/api/users` — List all users (auth required)
 - GET `/api/users/:email` — Get user by email
 - POST `/api/users/change-password` — Change password
- Authentication via `x-api-key` header.

Source: `k8s/components/n8n/user-manager/`

Manifests: `k8s/components/n8n/`

6.4 Ray Cluster (Distributed Computing)

Ray provides distributed computing for ML training and inference.

| Component | Node | GPU | Purpose |
|------------|----------|--------------------|-----------------------|
| ray-head | Chimera | None (coordinator) | Scheduling, dashboard |
| ray-worker | Cerberus | 2x RTX 5090 | Training compute |

Manifests: `k8s/components/ray/`

6.5 Other Services

| Service | Port | Description |
|---------------|-------|---|
| hackathons | 45821 | Hackathon voting/judging app (Vue.js + Express) |
| java-executor | 55392 | Remote Java code execution service |
| git-learning | 8080 | Interactive Git learning environment |
| sshpiper | 2222 | SSH proxy routing to student containers |

Manifests: `k8s/components/{hackathons,java-executor,git-learning,sshpiper}/`

7 GPU Infrastructure

7.1 NVIDIA GPU Operator

The GPU Operator runs in the `gpu-operator` namespace and manages:

- Device plugin (exposes GPUs to K8s scheduler)
- Container toolkit (`nvidia-container-runtime`)
- GPU Feature Discovery (node labels)
- DCGM Exporter (GPU metrics)
- CUDA validator (verifies GPU access)

Hydra Exclusion: Hydra (control plane) has no GPUs. All `nvidia.com/gpu.deploy.*` labels are set to `false` on Hydra to prevent GPU operator pods from scheduling there.

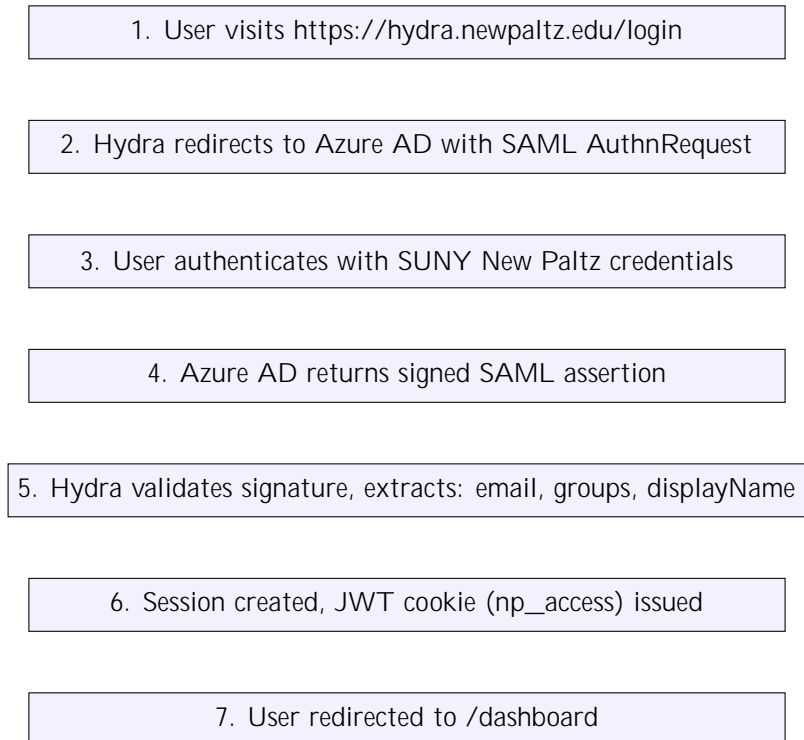
7.2 GPU Allocation

| Node | GPUs | Model | VRAM | Primary Consumer |
|----------|------|----------|-------------|-------------------------------------|
| Chimera | 3 | RTX 3090 | 72 GB total | Ollama (all 3) |
| Cerberus | 2 | RTX 5090 | 64 GB total | Ray Worker / Student GPU containers |

Part III

Authentication System

8 SAML 2.0 SSO Flow



9 Session and JWT Management

- **Express Session:** Server-side storage in SQLite
- **JWT Cookie (np_access):** Site-wide SSO cookie
- **JWKS Endpoint:** /.well-known/jwks.json for public key distribution
- **Algorithm:** RS256
- **TTL:** Configurable via JWT_TTL_SECONDS (default: 86400)
- **Cookie Domain:** .newpaltz.edu

10 Cross-Service Authentication

10.1 OpenWebUI Account Provisioning

When a user logs in via SAML, Hydra automatically provisions an OpenWebUI account via the middleman API. The password is derived and set transparently.

10.2 n8n Account Provisioning

Similarly, n8n accounts are provisioned via the n8n User Manager API on first login.

10.3 CS Lab JWT Verification

The CS Lab backend verifies JWT tokens using the public key mounted via ConfigMap `cs-lab-jwt-key` at `/app/server/keys/jwt-public.pem`.

Part IV

Student Containers

11 Container Features

Each student receives a persistent container with:

| Feature | Details |
|---------|--|
| Node.js | Latest LTS via nvm |
| Python | 3.11+ with pip, venv, Jupyter |
| Java | OpenJDK 21 |
| Docker | Docker CLI (Docker-in-Docker via rootless) |
| VS Code | code-server browser IDE (port 8443, always on) |
| Jupyter | JupyterLab (port 8888, approval required) |
| Jenkins | CI/CD server (port 8080, approval required) |
| SSH | Direct access via SSHPiper (port 2222) |
| Tools | Git, curl, wget, build-essential, gdb, cmake |

11.1 Service Management

Each student container runs managed services via `supervisord`. Code-server and Jenkins are always available; Jupyter requires admin approval.

| Service | Port | Path | Autostart | Approval |
|-------------|------|---------------------------|-----------|----------------------------|
| code-server | 8443 | /students/{user}/vscode/ | On init | None required |
| Jenkins | 8080 | /students/{user}/jenkins/ | On init | None required |
| Jupyter Lab | 8888 | /students/{user}/jupyter/ | No | jupyter_execution_approved |
| SSH | 22 | via SSHPiper port 2222 | Yes | None required |

Jupyter approval flow:

1. Student clicks "Request Jupyter Access" on the dashboard
2. Request stored in `resource_requests` table (type: `jupyter_execution`)
3. Admin approves via admin panel, setting `jupyter_execution_approved` in `user_quotas`
4. On next container init, the `JUPYTER_APPROVED` env var is injected
5. `entrypoint.sh` creates `/var/run/jupyter-approved` marker
6. Supervisor starts Jupyter

CLI gating: Jupyter also has a CLI wrapper (`jupyter-gate.sh`) that blocks direct `jupyter` command usage. The real binary is at `/usr/local/bin/jupyter.real`. Students see an error directing them to request access via the dashboard.

Jenkins: Always available to all students. Start/Stop/Open buttons on the dashboard. Jenkins data persists at `~/.jenkins` in the student's PVC. See the dashboard FAQ for Jenkinsfile examples and test commands.

12 Container Presets

- **Jupyter:** `jupyter/minimal-notebook`, port 8888, ForwardAuth
- **Static:** `nginx:alpine`, port 80, no auth
- **Repo:** Cloned from GitHub, runtime varies (Node/Python/nginx)
- **VS Code:** `codercom/code-server`, mounts any project volume

13 Resource Presets

| Preset | RAM | CPU | Storage | GPU | Approval |
|---------------|--------|-----|---------|-----|----------|
| Minimal | 256 MB | 0.5 | 5 GB | 0 | Auto |
| Conservative | 512 MB | 1 | 10 GB | 0 | Auto |
| Standard | 1 GB | 1 | 20 GB | 0 | Auto |
| Enhanced | 2 GB | 2 | 40 GB | 0 | Required |
| GPU Inference | 32 GB | 8 | 100 GB | 1 | Required |
| GPU Training | 48 GB | 16 | 200 GB | 2 | Required |

14 Pod Timing and Lifecycle

14.1 Duration Tiers

When requesting resources, students select a duration for how long their allocation lasts:

| Duration | Label | Auto-Approve | Description |
|----------|----------|--------------|------------------|
| 1 day | Default | Yes | Quick testing |
| 3 days | Short | Yes | Short assignment |
| 7 days | 1 Week | Yes | Short projects |
| 14 days | 2 Weeks | Yes | Standard project |
| 30 days | 1 Month | Yes | Semester project |
| 60 days | 2 Months | No | Extended project |
| 90 days | 3 Months | No | Full semester |

Config: `config/resources.js` lines 201–213. Default: 1 day. Maximum: 365 days (enforced at `routes/resource-requests.js:500`).

14.2 Resource Limits

| Limit | Value |
|--------------------------|--------|
| Max containers per user | 1 |
| Max storage per user | 200 GB |
| Max memory per container | 48 GB |
| Max CPUs per container | 16 |
| Max GPUs per container | 2 |

Config: `config/resources.js` lines 227–234.

14.3 Auto-Approval Thresholds

Requests within these limits are automatically approved without admin intervention:

| Resource | Auto-Approve Up To |
|----------|--------------------|
| Memory | 4 GB |
| CPUs | 2 cores |
| Storage | 40 GB |

Conservative presets on Hydra are always auto-approved. GPU requests always require admin approval. Pending requests expire after 7 days.

Config: `config/resources.js` lines 236–246.

14.4 K8s Resource Quotas per Preset

Each preset maps to specific Kubernetes requests/limits:

| Preset | Request Mem | Limit Mem | Request CPU | Limit CPU |
|---------------|-------------|-----------|-------------|-----------|
| Minimal | 512Mi | 1Gi | 250m | 1 |
| Conservative | 768Mi | 1536Mi | 500m | 1 |
| Standard | 1Gi | 2Gi | 500m | 2 |
| Enhanced | 2Gi | 4Gi | 1 | 4 |
| GPU Inference | 16Gi | 32Gi | 4 | 8 |
| GPU Training | 32Gi | 48Gi | 8 | 16 |

Config: `config/resources.js` lines 256–282.

14.5 Max Concurrent Pods per Node

| Node | Max Containers | Notes |
|----------|----------------|-----------------------------------|
| Hydra | 100 | No GPU, control plane |
| Chimera | 20 | 3 GPUs (1 reserved for OpenWebUI) |
| Cerberus | 10 | 2 GPUs (training) |

14.6 Pod Lifecycle Settings

| Setting | Value |
|--------------------------|---|
| Restart policy | Always |
| Termination grace period | 30 seconds |
| Resource requests | 50% of limits |
| Image pull policy | IfNotPresent |
| Security context | fsGroup: 1000, seccompProfile: RuntimeDefault |
| Service account | student-workload |
| PVC mount | /home/student (persists across restarts) |

14.7 JWT Session Duration

| Setting | Value |
|--------------------|-----------------------------|
| Default JWT TTL | 900 seconds (15 minutes) |
| Production JWT TTL | 2,592,000 seconds (30 days) |
| Algorithm | RS256 |

Configured via `JWT_TTL_SECONDS` in `.env`. JWKS endpoint: `/.well-known/jwks.json`.

15 SSH Access via SSHPiper

Students access containers via SSH through the SSHPiper proxy:

```
# Connect to your container
ssh -p 2222 student@hydra.newpaltz.edu

# Port 2222 is forwarded through the router to the sshpiper K8s pod
```

- SSHPiper routes connections based on username to the correct student pod
- Passwords displayed in dashboard after container creation
- Key-based auth supported (~/.ssh/authorized_keys)

16 Container Labels and Routing

Common labels on student containers:

- `hydra.managed_by=hydra-saml-auth`
- `hydra.owner=<username>`
- `hydra.project=<project>`
- `hydra.basePath=/students/<user>/<project>`

Traefik routes requests at `/students/<user>/<project>` to the corresponding container using StripPrefix middleware (except Jupyter, which uses `base_url`).

Part V

Networking

17 Firewall Configuration (UFW)

17.1 Hydra (Control Plane)

| | | | |
|---------------|-------|------------------------------|-------------------|
| 22/tcp | ALLOW | Anywhere | # SSH |
| 80/tcp | ALLOW | Anywhere | # HTTP |
| 443 | ALLOW | Anywhere | # HTTPS |
| 6969 | ALLOW | 172.17.0.0/16, 172.24.0.0/16 | # Auth (Docker) |
| 51820/udp | ALLOW | Anywhere | # WireGuard |
| 6443/tcp | ALLOW | 192.168.1.0/24 | # K8s API |
| 9345/tcp | ALLOW | 192.168.1.0/24 | # RKE2 supervisor |
| 10250/tcp | ALLOW | 192.168.1.0/24 | # Kubelet |
| 2379:2380/tcp | ALLOW | 192.168.1.0/24 | # etcd |
| 2222/tcp | ALLOW | Anywhere | # SSHPiper |
| 2049/tcp | ALLOW | 192.168.1.0/24 | # NFS |
| 111/tcp,udp | ALLOW | 192.168.1.0/24 | # portmapper |
| 8472/udp | ALLOW | 192.168.1.0/24 | # Flannel VXLAN |

17.2 Chimera (GPU Worker)

| | | | |
|-----------|-------|----------------|-----------------------|
| 22/tcp | ALLOW | Anywhere | # SSH |
| 7070/tcp | ALLOW | 192.168.1.148 | # OpenWebUI middleman |
| 9100 | ALLOW | 192.168.1.0/24 | # Metrics |
| 8472/udp | ALLOW | 192.168.1.0/24 | # Flannel VXLAN |
| 10250/tcp | ALLOW | 192.168.1.0/24 | # Kubelet |

17.3 Cerberus (GPU Worker)

| | | | |
|-----------|-------|----------------|----------------------|
| 22/tcp | ALLOW | Anywhere | # SSH |
| 9100 | ALLOW | 192.168.1.160 | # Metrics from Hydra |
| 2376 | ALLOW | 192.168.1.160 | # Docker from Hydra |
| 8472/udp | ALLOW | 192.168.1.0/24 | # Flannel VXLAN |
| 10250/tcp | ALLOW | 192.168.1.0/24 | # Kubelet |

18 Router Port Forwarding

| External Port | Internal IP | Internal Port | Service |
|---------------|---------------|---------------|------------------------|
| 22 | 192.168.1.160 | 22 | Admin SSH |
| 80 | 192.168.1.160 | 80 | HTTP |
| 443 | 192.168.1.160 | 443 | HTTPS |
| 2222 | 192.168.1.160 | 2222 | Student SSH (SSHPiper) |

19 DNS

- `hydra.newpaltz.edu` — Main domain, points to campus public IP

- `gpt.hydra.newpaltz.edu` — OpenWebUI subdomain
 - `n8n.hydra.newpaltz.edu` — n8n subdomain
- TLS certificates managed by Let's Encrypt via Traefik ACME.

Part VI

Deployment and Operations

20 Ansible Playbooks

The cluster can be deployed from scratch using Ansible playbooks in `ansible/`:

```
cd /home/infra/hydra-saml-auth/ansible
ansible-playbook -i inventory.yml playbooks/site.yml
```

20.1 Playbook Execution Order

1. `00-preflight-backup.yml` — Create backups before changes
2. `01-prepare-nodes.yml` — Install packages, configure kernel
3. `02-rke2-server.yml` — Setup RKE2 control plane on Hydra
4. `03-rke2-agents.yml` — Join Chimera and Cerberus to cluster
5. `04-gpu-setup.yml` — Configure NVIDIA drivers and GPU Operator
6. `05-deploy-hydra.yml` — Deploy all K8s manifests

20.2 What `05-deploy-hydra.yml` Deploys

In order:

1. Namespaces and RBAC
2. Storage classes
3. Traefik CRDs and deployment
4. Hydra Auth deployment
5. CS Lab website (backend + DB)
6. Hackathons, Java Executor, Git Learning
7. SSHPiper
8. n8n (app + Postgres + user manager)
9. Ollama
10. OpenWebUI (with middleman sidecar)
11. Ray cluster (head + worker)

21 CS Lab Website Deployment

```
# 1. Build the image
cd /home/infra/NewPaltz-CS-Lab-Website
docker build --no-cache -t newpaltz-cs-lab-website-backend:latest .

# 2. Export to tarball
docker save newpaltz-cs-lab-website-backend:latest \
  -o /data/containers/images/newpaltz-cs-lab-website-backend-latest.tar

# 3. Import into RKE2's containerd
sudo ctr --address /run/k3s/containerd/containerd.sock \
  -n k8s.io images import \
  /data/containers/images/newpaltz-cs-lab-website-backend-latest.tar
```

```
# 4. Restart the pod
kubectl delete pod -l app.kubernetes.io/component=backend -n hydra-system
```

Docker vs RKE2 Containerd: Docker and RKE2 use separate containerd instances with separate image stores. Docker builds go to Docker's containerd. You must explicitly import images into RKE2's containerd at `/run/k3s/containerd/containerd.sock`.

22 Image Management

- Image tarballs stored at `/data/containers/images/`
- Use `imagePullPolicy`: `Never` for locally-imported images
- Use unique tags (e.g., `v20260206144528`) to force pod recreation

23 Backup System

23.1 Daily Cluster Backups

| Setting | Value |
|-----------------|--|
| Backup Location | <code>/mnt/sdh4/backups/</code> |
| Schedule | Daily at 1:00 AM (crontab) |
| Method | rsync with compression |
| Script | <code>/home/infra/backup-cluster.sh</code> |
| Log File | <code>/var/log/cluster-backup.log</code> |

23.2 etcd Snapshots

Automatic every 12 hours via RKE2. Stored in `/var/lib/rancher/rke2/server/db/snapshots/`.

23.3 Backup Exclusions

`/dev/*, /proc/*, /sys/*, /run/*, /tmp/*, /var/tmp/*, /var/cache/*, /mnt/*, /var/lib/docker/*, /var/lib/rancher/*`

24 Automation and Scheduled Tasks

All recurring automation is documented here for operational reference.

24.1 System Cron Jobs

| Schedule | Component | Script | Purpose |
|--------------------|-----------------------------------|---|---|
| Daily 1:00 AM | Root crontab | <code>/usr/local/bin/backup-cluster.sh</code> | rsync backup of <code>/mnt/sdh4/backu</code> |
| Weekly Sat 2:45 AM | Root crontab | <code>certbot renew</code> | Let's Encrypt SSL newal |
| 1st Sunday/month | <code>/etc/cron.d/zfsutils</code> | <code>/usr/lib/zfs-linux/trim</code> | ZFS TRIM |
| 2nd Sunday/month | <code>/etc/cron.d/zfsutils</code> | <code>/usr/lib/zfs-linux/scrub</code> | ZFS scrub integri |
| Every 12 hours | RKE2 built-in | <code>etcd auto-snapshot</code> | Stored <code>/var/lib/ranche</code> |

24.2 Application Background Services

These services run inside the `hydra-auth` Node.js process (started in `index.js` lines 906–958):

| Service | Interval | File | Purpose |
|---------------------|------------|---|--|
| Metrics collector | 30 seconds | <code>services/metrics-collector.js</code> | Collects CPU/RAM/disk from all 3 nodes (Chimera/Cerberus via port 9100) |
| Security monitor | 5 minutes | <code>services/security-monitor.js</code> | Mining detection (18 known miners), CPU/RAM thresh- old alerts |
| Resource expiry | 1 hour | <code>services/resource-expiry.js</code> | Migrates expired GPU con- tainers back to Hydra, resets to defaults |
| Container reminders | 24 hours | <code>services/container-reminder.js</code> | Monthly email reminders to students about their contain- ers |

24.3 Security Monitor Thresholds

| Metric | Warning | Critical |
|-----------------|---------|----------------------|
| CPU usage | 80% | 95% |
| Memory usage | 85% | 95% |
| Mining detected | — | Auto-pause container |

Mining enforcement is controlled by `MINING_ENFORCEMENT_ENABLED` in `.env` (default: `true`).
Detects: `xmrig`, `ethminer`, `cgminer`, `nicehash`, etc. (18 process names).

Config: `SECURITY_STATS_INTERVAL` env var (default 300000ms = 5 min, set to 0 to disable).

24.4 Resource Expiry Behavior

When a student's GPU resource allocation expires:

1. Resource expiry checker detects `resources_expire_at` has passed (hourly check)
2. Container is migrated from GPU node (Chimera/Cerberus) back to Hydra
3. Resource config reset to defaults: 4 GB memory, 2 CPUs, 0 GPUs

4. Email notification sent to student
5. Database updated via `resetContainerConfigToDefaults()`

24.5 Dynamic Route Management

Traefik IngressRoutes and SSHPiper configs are managed dynamically:

| Event | Action | Details |
|-------------|-----------------|---|
| Pod init | Create routes | <code>k8sClient.createIngressRoute()</code> creates 3 routes (vscode, jupyter, jenkins) + ForwardAuth + Strip-Prefix middleware |
| Pod init | Update SSHPiper | Writes <code>sshpiper/config/{user}/sshpiper_upstream</code> with pod IP |
| Pod start | Update SSHPiper | Refreshes SSHPiper config with new pod IP |
| Pod destroy | Delete routes | <code>k8sClient.deleteIngressRoute()</code> and <code>deleteMiddleware()</code> |
| Pod destroy | Delete SSHPiper | Removes SSHPiper config directory for user |

Key files:

- Route creation: `services/k8s-containers.js` lines 245–323 (`buildIngressRouteSpec`, `buildMiddlewareSpec`)
- SSHPiper update: `services/k8s-containers.js` lines 415–431 (`updateSshPiperConfig`)
- Route recovery on boot: `scripts/fix-k8s-routes.sh` (systemd one-shot service)

24.6 Environment Variables for Automation

| Variable | Default | Purpose |
|---|---------|--|
| <code>SECURITY_STATS_INTERVAL</code> | 300000 | Security check interval (ms), 0 to disable |
| <code>MINING_ENFORCEMENT_ENABLED</code> | true | Auto-pause containers running miners |
| <code>JWT_TTL_SECONDS</code> | 900 | JWT token lifetime (production: 2592000) |
| <code>MAIL_METHOD</code> | — | Email backend: <code>graph</code> or <code>smtp</code> |
| <code>MS_TENANT_ID</code> | — | Azure AD tenant for Graph email API |
| <code>MS_CLIENT_ID</code> | — | Azure AD client ID |
| <code>MS_CLIENT_SECRET</code> | — | Azure AD client secret |

25 Common Operations

25.1 Kubectl Shortcuts

Sourced from `~/.hydra-aliases`:

```
k          # kubectl
kgp        # kubectl get pods -A
kgs        # kubectl get svc -A
students   # list student pods
hydra-health # quick cluster health check
gpu-check  # GPU availability per node
```

25.2 Service Management

```
# View all pods
kubectl get pods -A

# Restart a deployment
kubectl rollout restart deployment/<name> -n <namespace>

# View logs
kubectl logs -f deployment/<name> -n <namespace>

# Execute shell in pod
kubectl exec -it <pod-name> -n <namespace> -- /bin/bash
```

Part VII

Web Services and Route Map

This section documents every web-facing service, its URL, source code location, and how to update it.

26 Complete Site Inventory

| Service | URL | Port | Source Code |
|------------------------|--|-------|------------------------------------|
| Hydra Auth (Dashboard) | /dashboard, /login, /auth, /servers | 6969 | ~/hydra-saml-auth/ |
| CS Lab Website | / (catch-all), /courses, /events, /faculty | 5001 | ~/NewPaltz-CS-Lab- Website/ |
| Hackathons | /hackathons | 45821 | ~/Hackaton-Voting/ |
| Git Learning | /git | 38765 | ~/GG-git-learning/ |
| FLAPJS | /jflap | 8080 | ~/FLAPJS-WebApp/ |
| Java Executor | /java | 3000 | ~/java-executor/ |
| OpenWebUI | gpt.hydra.newpaltz.edu | 3000 | Pre-built image (ghcr.io) |
| n8n | n8n.hydra.newpaltz.edu | 5678 | Pre-built image (docker.n8n.io) |
| Student VS Code | /students/{user}/vscode/ | 8443 | In student container |
| Student Jupyter | /students/{user}/jupyter/ | 8888 | In student container |
| Student Jenkins | /students/{user}/jenkins/ | 8080 | In student container |

27 Traefik Route Priority Table

Route conflicts: If a new API path overlaps with an existing route (e.g., both hydra-auth and cs-lab use `/api/events`), the higher-priority route wins. Always check existing routes before adding new ones: `kubectl get ingressroute -A -o wide`.

28 How to Update Each Service

28.1 Hydra Auth (Dashboard)

```
# Build and deploy using the build-deploy script:
cd ~/hydra-saml-auth
./scripts/build-deploy.sh auth

# Or manually:
sudo buildah bud -t ndg8743/hydra-saml-auth:vNEW .
sudo buildah push ndg8743/hydra-saml-auth:vNEW \
  docker-archive:/tmp/hydra-auth.tar:docker.io/ndg8743/hydra-saml-auth:
  vNEW
sudo ctr --address /run/k3s/containerd/containerd.sock \
  -n k8s.io images import /tmp/hydra-auth.tar
kubectl -n hydra-system set image deploy/hydra-auth \
  hydra-auth=docker.io/ndg8743/hydra-saml-auth:vNEW
```

Key files: `index.js` (auth), `routes/containers.js` (container API), `views/dashboard.ejs` (frontend), `config/resources.js` (presets).

28.2 CS Lab Website

```
cd ~/NewPaltz-CS-Lab-Website
sudo buildah bud -t newpaltz-cs-lab-website-backend:latest .
sudo rm -f /tmp/cs-lab.tar
sudo buildah push newpaltz-cs-lab-website-backend:latest \
  docker-archive:/tmp/cs-lab.tar:docker.io/newpaltz-cs-lab-website-
  backend:latest
sudo ctr --address /run/k3s/containerd/containerd.sock \
  -n k8s.io images import /tmp/cs-lab.tar
kubectl -n hydra-system rollout restart deploy/cs-lab
```

Stack: Node.js + Express backend, Vue.js/Vite frontend. Database: SQLite.

28.3 Hackathons

```
cd ~/Hackaton-Voting
sudo buildah bud -t hackaton-voting-app:latest .
# Export, import, restart same as above
kubectl -n hydra-infra rollout restart deploy/hackathons
```

Stack: Vue.js frontend, Express backend, SQLite.

28.4 FLAPJS

```
cd ~/FLAPJS-WebApp
sudo buildah bud -t flapjs-webapp:latest .
# Export, import, restart
kubectl -n hydra-infra rollout restart deploy/flapjs
```

Stack: React SPA built with webpack, served by Nginx. Uses `sub_filter` to inject `<base href="/jflap/">` for subpath routing.

28.5 Git Learning

```
cd ~/GG-git-learning
sudo buildah bud -t gg-git-learning-app:latest .
kubectl -n hydra-infra rollout restart deploy/git-learning
```

Stack: Node.js with PM2 runtime.

28.6 Java Executor

```
cd ~/java-executor
sudo buildah bud -t docker-java-executor-java-executor:latest .
kubectl -n hydra-infra rollout restart deploy/java-executor
```

Note: Mounts host Docker socket for container-based Java compilation.

28.7 Student Container Image

```
cd ~/hydra-saml-auth
./scripts/build-deploy.sh student

# This builds the student-container image and notes that
# existing student pods need restart to use the new image.
# Update STUDENT_IMAGE in .env if using a versioned tag.
```

28.8 OpenWebUI and n8n

These use upstream pre-built images. To update:

```
# OpenWebUI: Update image tag in deployment spec
kubectl -n hydra-infra set image deploy/open-webui \
  open-webui=ghcr.io/open-webui/open-webui:vNEW

# n8n: Update image tag in deployment spec
kubectl -n hydra-infra set image deploy/n8n \
  n8n=docker.n8n.io/n8nio/n8n:NEW_VERSION
```

29 K8s Manifests Location

All K8s deployment manifests live in `~/hydra-saml-auth/k8s/`:

| Service | Manifest Path |
|---------------------------|-------------------------------|
| Namespaces, RBAC, storage | k8s/base/ |
| Hydra Auth | k8s/components/hydra-auth/ |
| CS Lab | k8s/components/cs-lab/ |
| Traefik | k8s/components/traefik/ |
| Hackathons | k8s/components/hackathons/ |
| Git Learning | k8s/components/git-learning/ |
| FLAPJS | k8s/components/flapjs/ |
| Java Executor | k8s/components/java-executor/ |
| n8n | k8s/components/n8n/ |
| OpenWebUI | k8s/components/openwebui/ |
| Ollama | k8s/components/ollama/ |
| Student Pods | k8s/components/student-pods/ |
| SSHPiper | k8s/components/sshpiper/ |

30 Namespace Layout

| Namespace | Contents |
|----------------|---|
| hydra-system | Core services: traefik, hydra-auth, cs-lab |
| hydra-infra | Infrastructure: n8n, openwebui, ollama, ray, git-learning, hackathons, flapjs, java-executor, ssh-piper |
| hydra-students | Student containers (dynamically created per-user) |
| gpu-operator | NVIDIA GPU Operator, device plugin, DCGM exporter |
| kube-system | K8s system components (etcd, coredns, canal, metrics-server) |

Part VIII

OpenWebUI API Integration

31 Getting Started

1. Log in at <https://hydra.newpaltz.edu/dashboard>
2. Visit <https://gpt.hydra.newpaltz.edu>
3. Go to Settings → Account → Generate New API Key
4. Copy the key (format: sk-...) — shown only once

32 API Configuration

```
ENDPOINT=https://gpt.hydra.newpaltz.edu/api/chat/completions
MODEL=gemma3:12b
API_KEY=sk-your-api-key-here
```

33 cURL Example

```
curl https://gpt.hydra.newpaltz.edu/api/chat/completions \
-H "Content-Type: application/json" \
-H "Authorization: Bearer sk-your-api-key-here" \
-d '{
  "model": "gemma3:12b",
  "messages": [{"role": "user", "content": "Hello!"}]
}'
```

34 Python Example

```
import openai, os
openai.api_base = "https://gpt.hydra.newpaltz.edu/api"
openai.api_key = os.getenv("HYDRA_API_KEY")

response = openai.ChatCompletion.create(
    model="gemma3:12b",
    messages=[{"role": "user", "content": "Hello!"}]
)
print(response.choices[0].message.content)
```

35 JavaScript Example

```
const response = await fetch(
  'https://gpt.hydra.newpaltz.edu/api/chat/completions',
  {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
```

```
    'Authorization': 'Bearer ' + API_KEY
  },
  body: JSON.stringify({
    model: 'gemma3:12b',
    messages: [{role: 'user', content: 'Hello!'}]
  })
}
);
const data = await response.json();
console.log(data.choices[0].message.content);
```

Additional language examples (PHP, Java, C#, Ruby, Go, Rust) are available in the full API access guide at [docs/access.md](#).

Part IX

Security

36 Security Architecture Layers

1. **Network:** UFW firewall, TLS encryption, CORS policy
2. **Authentication:** SAML 2.0, JWT tokens, API keys
3. **Authorization:** Role-based access, container ownership labels
4. **Runtime:** Container isolation, resource limits, seccomp profiles

37 Known Vulnerabilities

Critical:

- **Privileged containers** in Docker mode grant full host access
- **Docker socket mount** in student containers is equivalent to root on host

High:

- Passwordless sudo for student user in container images
- Supervisor web interface (port 9001) without authentication
- Mining detection without automatic enforcement
- K8s pod security context missing `runAsNonRoot`, `allowPrivilegeEscalation: false`

Medium:

- No NetworkPolicy isolation between student pods
- No PID limits (fork bomb vulnerability)
- Jupyter/VS Code without application-level auth (relies on ForwardAuth)

See `docs/SECURITY_VULNERABILITIES.md` for full details and remediation steps.

38 Security Best Practices for Students

- Never share API keys publicly or commit to version control
- Use environment variables for sensitive configuration
- Rotate API keys regularly
- Use HTTPS only for all API communications
- Validate and sanitize user inputs before sending to API

Part X

RDMA and GPUDirect

39 Overview

The cluster supports RDMA networking for high-performance GPU-to-GPU communication:

| Node | NIC | GPUs | RDMA |
|----------|----------|-------------|---------------------------|
| Hydra | Onboard | None | SoftRoCE (testing) |
| Chimera | ConnectX | 3x RTX 3090 | Hardware RoCE |
| Cerberus | ConnectX | 2x RTX 5090 | Hardware RoCE + GPUDirect |

40 Installation Order (Critical)

1. MLNX_OFED / DOCA (network drivers)
2. NVIDIA GPU Drivers (includes nvidia-peermem)
3. CUDA Toolkit
4. Load nvidia-peermem module

If the NVIDIA GPU driver is installed before MLNX_OFED, the driver must be reinstalled to compile nvidia-peermem with RDMA APIs.

41 SoftRoCE Setup

```
# Install prerequisites
sudo apt install rdma-core ibverbs-utils perftest

# Create SoftRoCE device
sudo rdma link add rxe0 type rxe netdev eth0

# Verify
rdma link && ibv_devices
```

42 GPUDirect RDMA Verification

```
# Load nvidia-peermem
sudo modprobe nvidia-peermem

# Make persistent
echo "nvidia-peermem" | sudo tee /etc/modules-load.d/nvidia-peermem.conf

# Test bandwidth (two nodes)
# Server: ib_write_bw -d mlx5_0 --use_cuda=0
# Client: ib_write_bw -d mlx5_0 --use_cuda=0 <server_ip>
```

See `docs/rdma-gpudirect-setup.md` for complete SR-IOV, DOCA, and KVM passthrough configuration.

Part XI

Troubleshooting

43 Authentication Issues

| Symptom | Solution |
|------------------------|--|
| SAML assertion invalid | Verify <code>METADATA_URL</code> and <code>SAML_SP_ENTITY_ID</code> match Azure config |
| Cookie not set | Check <code>COOKIE_DOMAIN</code> , ensure HTTPS |
| JWT verification fails | Check JWKS endpoint accessible, verify key rotation |

44 Container Issues

| Symptom | Solution |
|-----------------------|--|
| Container won't start | Verify student container image exists in RKE2 containerd |
| Container 404 | Check Traefik is running, container has correct labels |
| VS Code not loading | Check code-server process, ForwardAuth middleware |
| Jupyter issues | Verify <code>base_url</code> setting matches path |
| SSH not working | Check SSHPiper pod, verify port 2222 routing |
| Files not persisting | Only <code>/home/student/</code> is persisted via PVC |

45 GPU Issues

| Symptom | Solution |
|--------------------------|---|
| GPU not detected | Run <code>nvidia-smi</code> on host, check NVIDIA drivers |
| GPU pod pending | Check GPU operator pods in gpu-operator namespace |
| Ollama can't use GPU | Verify all 3 GPUs allocated to Ollama deployment |
| Ray worker can't use GPU | Check NVIDIA device plugin on Cerberus |

46 Networking Issues

| Symptom | Solution |
|-----------------------|--|
| Service unreachable | Check pod is Running, service exists, Ingress-Route matches |
| 502 Bad Gateway | Backend pod crashed or port mismatch |
| TLS certificate error | Check Traefik ACME, run <code>certbot renew --dry-run</code> |
| NFS mount failed | Verify NFS server running on Hydra, firewall allows 2049 |
| Cross-node pod issue | Check Flannel VXLAN (8472/udp) allowed between nodes |

47 Traefik Deployment Issues

Stuck Rolling Update: Traefik uses `hostPort` which means only one pod can bind ports 80/443 at a time. The deployment MUST use `strategy: Recreate` (not `RollingUpdate`). If stuck:

```
kubectl rollout undo deployment/traefik -n hydra-system
```

48 CS Lab Website Catch-All Route

The Express server has a catch-all that serves `index.html` for SPA routes. Backend API paths are excluded:

```
# Paths excluded from SPA catch-all (served by backend):
/api/*, /faq, /faculty, /uploads, /scripts, /tech-blog,
/student-resources, /student-highlights, /admins, /auth,
/school-calendar, /sd-forms

# Paths explicitly allowed through for SPA routing:
/student-forms, /submit-*
```

If adding new frontend routes starting with `/student`, update the catch-all in `server.js`.

Part XII

Repository Structure

49 hydra-saml-auth

```

hydra-saml-auth/
|-- index.js                # SAML auth, JWT, routes, WebSocket
|-- routes/
|   |-- containers.js       # Container lifecycle
|   |-- resource-requests.js # Resource allocations
|   |-- webui-api.js        # OpenWebUI proxy
|   |-- n8n-api.js          # n8n account management
|   |-- servers-api.js      # Cluster status
|   |-- admin.js            # Admin panel
|-- services/
|   |-- db-init.js          # Database init
|   |-- resource-expiry.js   # Resource expiry checker
|   |-- security-monitor.js  # Process monitoring
|-- config/
|   |-- resources.js         # Presets and node config
|   |-- runtime.js           # Docker/K8s switcher
|-- k8s/
|   |-- base/               # Namespace, RBAC, storage
|   |-- components/
|   |   |-- traefik/        # Reverse proxy
|   |   |-- hydra-auth/     # Auth service
|   |   |-- cs-lab/         # CS Lab website
|   |   |-- ollama/         # LLM inference
|   |   |-- openwebui/      # AI chat + middleman
|   |   |-- n8n/            # Workflows + user manager
|   |   |-- ray/            # Distributed computing
|   |   |-- hackathons/     # Hackathon app
|   |   |-- java-executor/  # Code execution
|   |   |-- git-learning/   # Git learning
|   |   |-- sshpiper/       # SSH proxy
|   |   |-- student-pods/   # Pod templates
|   |-- gpu/               # GPU operator config
|-- ansible/
|   |-- inventory.yml       # Node definitions
|   |-- playbooks/         # Deployment scripts
|-- student-container/
|   |-- Dockerfile          # Student image
|   |-- supervisord.conf    # Process manager
|-- docs/                   # This document + sources
|-- docker-compose.yaml     # Legacy Docker deployment

```

50 Other Repositories

| Repo | Path | Description |
|-------------------------|----------------------------|---------------------------------|
| NewPaltz-CS-Lab-Website | ~/NewPaltz-CS-Lab-Website/ | React + Express CS Lab homepage |
| Hackaton-Voting | ~/Hackaton-Voting/ | Vue.js hackathon app |

Part XIII

Environment Configuration

51 Required Variables (hydra-saml-auth)

| Variable | Description |
|-------------------|----------------------------------|
| BASE_URL | https://hydra.newpaltz.edu |
| METADATA_URL | Azure AD federation metadata URL |
| SAML_SP_ENTITY_ID | SP Entity ID (must match Azure) |
| COOKIE_DOMAIN | .newpaltz.edu |
| PORT | Service port (default: 6969) |
| DB_PATH | SQLite path (/app/data/hydra.db) |
| JWT_TTL_SECONDS | Token lifetime (default: 86400) |

52 Ansible Inventory Variables

```
rke2_version: "v1.28.4+rke2r1"
cluster_domain: hydra.newpaltz.edu
nfs_server: "192.168.1.160"
nfs_path: "/srv/hydra-nfs"
```

Appendices

A Cleanup History (February 2026)

A comprehensive infrastructure cleanup was performed February 4–7, 2026:

| Node | Action | Reclaimed |
|--------------|---|----------------|
| Hydra | Docker system prune | 114.8 GB |
| Hydra | Remove stale files (<code>/opt/local-path-provisioner.bak</code> , temp files) | 20+ GB |
| Hydra | Truncate backup log | 389 MB |
| Chimera | Remove Docker Ollama duplicate + prune | 41.2 GB |
| Cerberus | Docker system prune | 51.3 GB |
| Total | | ~227 GB |

Key cleanup actions:

- Migrated all services from Docker containers to K8s pods
- Archived `legacy/` directory to `legacy-archive` git branch
- Relocated middleman sources to `k8s/components/` directories
- Removed stale Apache configs, scripts, temp files across all nodes
- Cleaned orphaned Docker networks, volumes, and images
- Fixed Traefik stuck rolling update (added `strategy: Recreate`)
- Fixed Ray cluster (removed GPU request from head, deployed properly)
- Verified all middleman APIs operational
- Cloned `hydra-saml-auth` repo to all 3 nodes

B Migration History

The infrastructure evolved through several phases:

1. **Bare metal** — Apache web server, manual user management
2. **Docker Compose** — Containerized services, Nginx reverse proxy
3. **K3s** — Initial Kubernetes, migrated from Docker Compose
4. **RKE2** — Current production cluster (January 2026), Traefik ingress
5. **Infrastructure Overhaul** — February 9, 2026 (see below)

C February 9, 2026 — Infrastructure Overhaul

Following a 5-hour OOM death spiral that made the server unresponsive, the following changes were applied:

1. **Phase 1: Docker cleanup** — Pruned 11.86 GB orphaned Docker volumes, stopped and disabled Docker daemon on all nodes. Build tool changed to `buildah` (daemonless).
2. **Phase 2: Networking** — Removed nginx (conflicted with Traefik on port 80), fixed `SUNYCAT.png` route, fixed OpenWebUI cross-namespace reference, corrected OpenWebUI API fallback URL.
3. **Phase 3: CS Lab consolidation** — Removed MariaDB pod (app uses SQLite). Went from 2 pods (backend + MariaDB) to 1 pod (cs-lab). Removed `mariadb` npm dependency.
4. **Phase 4: RKE2 data migration** — Moved `/var/lib/rancher/rke2` (40 GB) to RAID at `/data/rke2`. Symlinked old path for backward compatibility. Config updated: `data-dir: /data/rke2`.

5. **Phase 5: OOM prevention** — Added 32 GB swap on RAID (`vm.swappiness=10`), kubelet eviction thresholds (`memory.available<2Gi` hard, 4Gi soft), `system-reserved=4Gi`, `kube-reserved=2Gi`. Applied `ResourceQuota` and `LimitRange` to `hydra-students` namespace (default 2Gi/1CPU per container, max 48Gi/16CPU). Enabled auto-reboot after kernel updates at 04:00.

Build tools after overhaul:

- `buildah` — Daemonless OCI image builder (primary)
- `nerdctl` — Docker-compatible CLI for containerd (`/usr/local/bin/nerdctl`)
- Docker daemon is **disabled** (`systemctl disable docker`)

D February 9, 2026 — Jenkins Service + Jupyter Gating + Repo Cleanup

1. **Jupyter execution gating** — Supervisor `autostart=false`, CLI gate wrapper (`jupyter-gate.sh`), API approval endpoints, `JUPYTER_APPROVED` env var marker. Students cannot run Jupyter until admin approves.
2. **Jenkins CI/CD service** — Added as 3rd managed service inside student containers. Supervisor config (`port 8080`, `autostart=false`), K8s pod/service/IngressRoute/strip-prefix routing, DB schema (`jenkins_execution_approved`), admin approval flow, dashboard UI card with Start/Stop/Open buttons.
3. **Repo cleanup** — Removed 12 dead Docker-era files (deploy scripts, old Python metrics agent, Apache config, SSHPiper Docker Compose, student-mvp, Ray reference compose files). Scrubbed plaintext secret files from git history. Added `.example` templates for K8s secrets.
4. **K8s template updates** — Added Jenkins port 8080 to pod template, Jenkins route/middleware/service port to IngressRoute template, flapjs deployment.
5. **Security** — Removed `cs-lab/secret.yaml` and `n8n/secret.yaml` from git tracking (contained plaintext credentials). Added to `.gitignore`. Credentials should be rotated.

E February 9, 2026 — Route Fixes + Pod Restart + Documentation

1. **FLAPJS fix** — Dockerfile wasn't copying `index.html` from project root (webpack outputs it outside `dist/`). Fixed Dockerfile + added `nginx sub_filter` for `<base href="/jflap/">` injection.
2. **Hydra-auth deployment** — Containerd `'latest'` tag was resolving to stale Docker Hub image. Switched to unique versioned tags (`v20260209-HHMMSS`) with `imagePullPolicy: Never`. Created `scripts/build-deploy.sh` for reliable single-path builds.
3. **Jenkins ungated** — Removed approval requirement for Jenkins. All students can now Start/Stop Jenkins from dashboard without admin approval.
4. **Traefik route conflict** — `/api/events` was being intercepted by `cs-lab-website` (priority 20) instead of `hydra-auth` (priority 15). Bumped `hydra-auth`'s route to priority 25.
5. **Pod batch restart** — All 26 student pods restarted from Completed state. PVC data preserved. Code-server started on all pods via `supervisorctl batch` command.
6. **Temp cleanup** — Freed 2.6 GB on Hydra (`/tmp` build artifacts), cleaned audit exports on Chimera and Cerberus.
7. **Documentation** — Added Pod Timing & Lifecycle section, Automation & Scheduled Tasks section, Web Services & Route Map with rebuild instructions for all services.

F References

- RKE2 Documentation: <https://docs.rke2.io/>
- Traefik Documentation: <https://doc.traefik.io/traefik/>
- SAML 2.0 Spec: <https://docs.oasis-open.org/security/saml/v2.0/>
- Azure AD SAML: <https://learn.microsoft.com/en-us/entra/identity/>
- NVIDIA GPU Operator: <https://docs.nvidia.com/datacenter/cloud-native/gpu-operator/>
- OpenWebUI: <https://docs.openwebui.com>
- Ray: <https://docs.ray.io/>
- n8n: <https://docs.n8n.io/>