Hydra Infrastructure Management Guide

Student Container Platform Administration

Computer Science Department
SUNY New Paltz

Last Updated: January 2025

Contents

1 System Overview

Hydra is a containerized development platform providing persistent development environments for Computer Science students and faculty at SUNY New Paltz. The system uses SAML 2.0 Single Sign-On via Azure AD and Docker for container orchestration.

1.1 Key Features

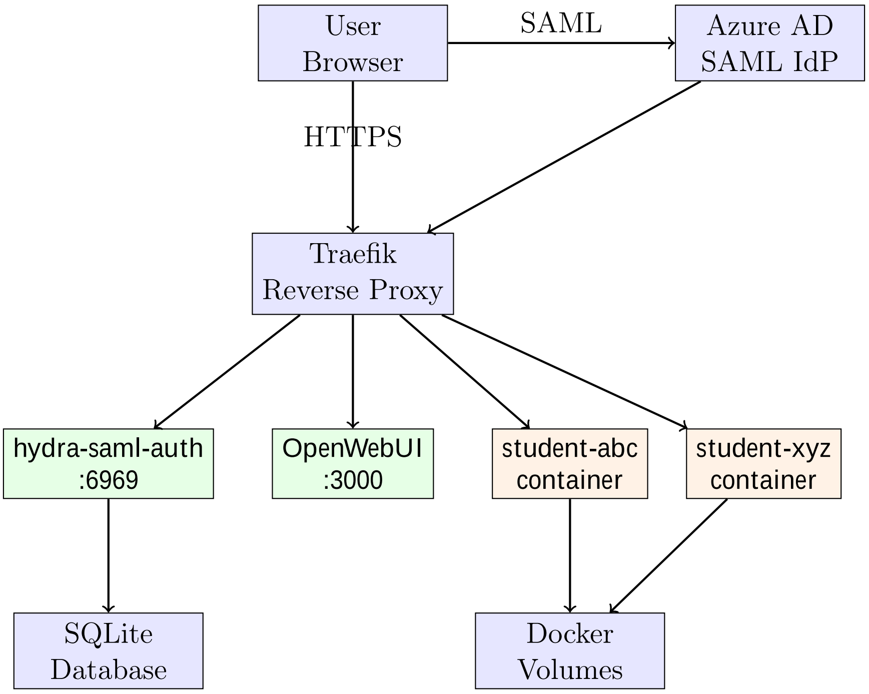
- **SSO Authentication:** Azure AD SAML 2.0 with automatic user provisioning
- **Persistent Containers:** One development environment per student with data persistence
- **Built-in Services:** VS Code (code-server), Jupyter Notebook, Docker-in-Docker
- **Dynamic Routing:** Traefik-based routing for custom web applications
- **Resource Management:** Per-container CPU and memory limits
- **Integration:** OpenWebUI (GPT) and n8n account management

1.2 Access URLs

Service	URL	Description
Dashboard	`https://hydra.newpaltz.edu/dashboard`	Main user interface
OpenWebUI	`https://gpt.hydra.newpaltz.edu/`	AI chat interface
VS Code	`https://hydra.newpaltz.edu/students/{user}/vscode`	Browser IDE
Jupyter	`https://hydra.newpaltz.edu/students/{user}/jupyter`	Notebooks

2 System Architecture

2.1 Architecture Diagram



2.2 Component Overview

Component	Port	Description
Traefik	80, 443	Reverse proxy, TLS termination, routing
hydra-saml-auth	6969	SAML auth, dashboard, container management
OpenWebUI	3000	AI chat interface (Ollama frontend)
Student Containers	Dynamic	Per-user development environments

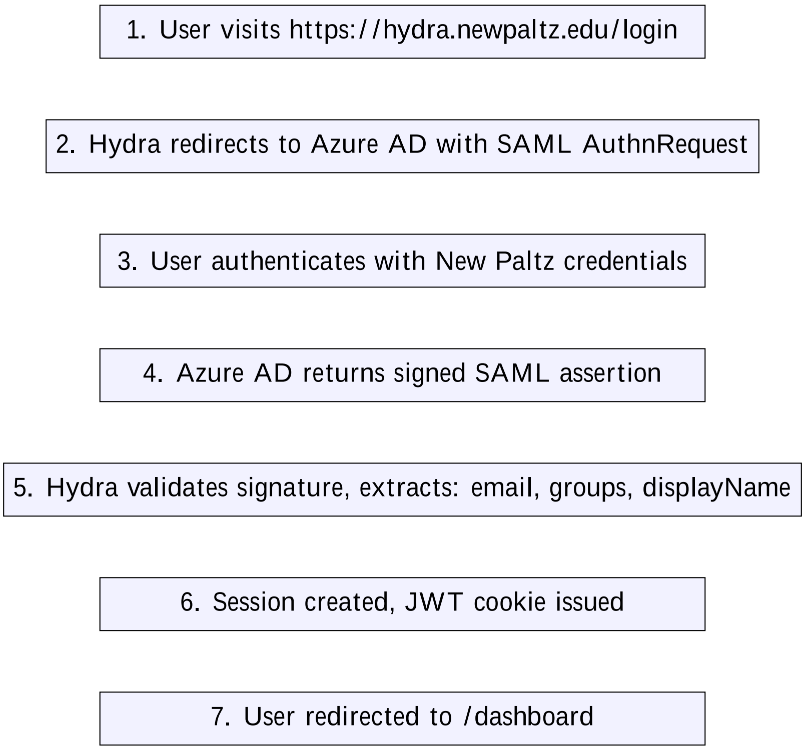
2.3 Network Architecture

Student containers operate on an isolated Docker network (`hydra_students_net`) with:

- No direct internet access (configurable)
- Internal DNS resolution
- Traefik-mediated external access via ForwardAuth

3 Authentication System

3.1 SAML 2.0 SSO Flow



1. User visits https://hydra.newpaltz.edu/login

2. Hydra redirects to Azure AD with SAML AuthnRequest

3. User authenticates with New Paltz credentials

4. Azure AD returns signed SAML assertion

5. Hydra validates signature, extracts: email, groups, displayName

6. Session created, JWT cookie issued

7. User redirected to /dashboard

3.2 Session Management

Sessions are managed via:

- **Express Session:** Server-side session storage in SQLite
- **JWT Cookie:** Site-wide authentication cookie for cross-service SSO

- **JWKS Endpoint:** Public key endpoint for JWT verification by other services

JWT Configuration:

- TTL: Configurable via `JWT_TTL_SECONDS` (default: 86400)
- Algorithm: RS256
- Cookie Domain: `.newpaltz.edu`

4 Container System

4.1 Student Container Features

Each student receives a single persistent container with:

Feature	Details
Node.js	Latest LTS via nvm
Python	3.11+ with pip, venv, Jupyter
Java	OpenJDK 21
Docker	Full Docker-in-Docker support (privileged mode)
VS Code	code-server browser IDE
Jupyter	Notebook and JupyterLab
Tools	Git, curl, wget, build-essential, etc.

4.2 Resource Limits

Resource	Limit
RAM	4GB per container
CPU	2 cores per container
Storage	Unlimited (host disk limited)

Security Note: Student containers run in privileged mode to support Docker-in-Docker. This grants elevated access. Monitor for abuse and consider disabling for untrusted users.

4.3 Port Routing

Students can expose web applications through custom routes:

- Default routes: `/students/{username}/vscode`, `/students/{username}/jupyter`
- Custom routes added via dashboard UI
- Reserved ports: 8443 (code-server), 8888 (Jupyter)
- All routes protected by ForwardAuth

5 File Structure

```
hydra-saml-auth/
|-- index.js                # Main entry: SAML, JWT/JWKS, routes,
   WebSocket
|-- db.js                   # SQLite database initialization
|-- routes/
|   |-- containers.js       # Container lifecycle, services, ports, logs
|   |-- webui-api.js        # OpenWebUI account proxy
|   |-- n8n-api.js          # n8n account management
|   |-- servers-api.js      # Cluster status endpoints
|   |-- admin.js            # Admin panel routes
|-- services/
|   |-- activity-logger.js    # Activity tracking
|   |-- email-notifications.js # Email alerts
|-- views/                  # EJS templates
|-- student-container/
|   |-- Dockerfile          # Ubuntu 22.04 + dev tools
|   |-- supervisord.conf    # Process manager config
|   |-- entrypoint.sh       # Container startup
|-- docker-compose.yaml     # Production stack
|-- docs/                   # Documentation
```

6 Common Operations

6.1 View Running Containers

```
docker ps --filter "name=student-"
```

6.2 Access Container Shell

```
docker exec -it student-<username> /bin/bash
```

6.3 View Container Logs

```
docker logs -f student-<username> --tail=100
```

6.4 Restart a Container

```
docker restart student-<username>
```

6.5 Remove a Stuck Container

```
docker rm -f student-<username>
```

6.6 Rebuild Student Container Image

```
cd student-container
docker build -t hydra-student-container:latest .
```

Note: Students with existing containers must recreate them to use updated images.

7 Service Management

7.1 Restart Main Service

```
docker compose restart hydra-saml-auth
```

7.2 Rebuild and Redeploy

```
docker compose build hydra-saml-auth
docker compose up -d hydra-saml-auth
```

7.3 View Service Logs

```
docker compose logs -f hydra-saml-auth
```

7.4 Check Traefik Routing

```
docker compose logs traefik | grep -i error
curl -I https://hydra.newpaltz.edu/
```

8 Troubleshooting

8.1 Authentication Issues

Symptom	Solution
SAML assertion invalid	Verify `METADATA_URL` and `SAML_SP_ENTITY_ID` match Azure config exactly
Cookie not set	Check `COOKIE_DOMAIN`, ensure HTTPS, check browser settings
JWT verification fails	Verify JWKS endpoint accessible, check key rotation

8.2 Container Issues

Symptom	Solution
Container won’t initialize	Verify `hydra-student-container:latest` image exists
Container 404	Check container is on `hydra_students_net`, Traefik running
Service won’t start	Check supervisord logs inside container
Port routing fails	Verify port not reserved (8443, 8888) and not in use

8.3 Service-Specific Issues

- **VS Code not loading:** Check code-server process, ForwardAuth working
- **Jupyter issues:** Verify `NotebookApp.base_url` setting
- **Docker-in-Docker fails:** Container must have privileged mode
- **Files not persisting:** Only `/home/student/` is persisted

9 Backup and Recovery

9.1 Database Backup

```
# Backup SQLite database
sqlite3 /app/data/webui.db ".backup ’/backups/hydra-$(date +%Y%m%d).db
   ’"

# Automated daily backup (add to crontab)
0 2 * * * sqlite3 /app/data/webui.db ".backup ’/backups/hydra-$(date
   +\%Y\%m\%d).db’"
```

9.2 Volume Backup

```
# List student volumes
docker volume ls | grep student-

# Backup a volume
docker run --rm -v student-<user>-data:/data -v $(pwd):/backup \
    alpine tar cvf /backup/student-<user>-backup.tar /data
```

10 Environment Configuration

10.1 Required Variables

Variable	Description
BASE_URL	External URL (https://hydra.newpaltz.edu)
METADATA_URL	Azure AD federation metadata URL
SAML_SP_ENTITY_ID	SP Entity ID (must match Azure exactly)
COOKIE_DOMAIN	Cookie scope (.newpaltz.edu)
PORT	Service port (default: 6969)
DB_PATH	Database path (/app/data/webui.db)

10.2 Optional Variables

Variable	Description
PUBLIC_STUDENTS_BASE	Student URL base (https://hydra.newpaltz.edu/students)
JWT_TTL_SECONDS	JWT token lifetime
JWT_PRIVATE_KEY_FILE	Path to JWT signing key
JWT_PUBLIC_KEY_FILE	Path to JWT verification key

11 References

- Docker Documentation: https://docs.docker.com/
- Traefik Documentation: https://doc.traefik.io/traefik/
- SAML 2.0 Specification: https://docs.oasis-open.org/security/saml/v2.0/
- Azure AD SAML: https://docs.microsoft.com/en-us/azure/active-directory/develop/single-sign-on-saml-protocol
- code-server: https://coder.com/docs/code-server/latest
- Jupyter: https://jupyter.org/documentation