Hydra Installation & Setup Guide

For System Administrators

Computer Science Department
SUNY New Paltz

Last Updated: January 2025

Contents

1 Prerequisites

1.1 System Requirements

Component	Requirement
Op rating Syst m	Ubuntu 22.04 LTS (r comm nd d)
Dock r	24.0+ with Compos V2
RAM	Minimum 8GB (16GB+ r comm nd d)
Storag	100GB+ SSD
N twork	Static IP, ports 80/443 op n

1.2 Required Access

- Azur AD admin acc ss for SAML Ent rpris Application s tup
- DNS control for `hydr .yourdom in.edu`
- TLS c rtificat (L t's Encrypt or institutional c rt)
- SSH acc ss to th host s rv r

1.3 Software Dependencies

```
# Install Docker
curl -fsSL https://get.docker.com | sh
sudo usermod -aG docker $USER

# Verify Docker Compose V2
docker compose version
# Should show: Docker Compose version v2.x.x

# Install useful tools
sudo apt install -y git curl jq sqlite3
```

2 Installation Steps

2.1 Step 1: Clone Repository

```
git clone https://github.com/your-org/hydra-saml-auth.git
```

2.3 Step 3: Configure Environment

Create a `.env` file in the project root:

```
# === Core Settings ===
PORT=6969
BASE_URL=https://hydra.yourdomain.edu
COOKIE_DOMAIN=.yourdomain.edu

# === SAML Configuration ===
METADATA_URL=https://login.microsoftonline.com/YOUR_TENANT/
   federationmetadata/2007-06/federationmetadata.xml
SAML_SP_ENTITY_ID=hydra-auth
SAML_CALLBACK_URL=https://hydra.yourdomain.edu/auth/callback

# === Database ===
DB_PATH=/app/data/webui.db

# === JWT Settings ===
JWT_TTL_SECONDS=86400
JWT_KEY_ID=hydra-key-1
JWT_PRIVATE_KEY_FILE=/app/certs/jwt-private.pem
JWT_PUBLIC_KEY_FILE=/app/certs/jwt-public.pem

# === Student Containers ===
PUBLIC_STUDENTS_BASE=https://hydra.yourdomain.edu/students
```

2.4 Step 4: Generate JWT Keys

```
mkdir -p certs
openssl genrsa -out certs/jwt-private.pem 2048
openssl rsa -in certs/jwt-private.pem -pubout -out certs/jwt-public.pem
```

2.5 Step 5: Configure Azure AD

1. Go to Azure Portal > Azure Active Directory > Enterprise Applications

2. Click "New application" > "Create your own application"

3. Name: "Hydra Auth", select "Non-gallery application"

4. Go to Single sign-on > SAML

5. Set Identifier (Entity ID): hydra-auth

6. Set Reply URL: https://hydra.yourdomain.edu/auth/callback

7. Download Federation Metadata XML, note the URL

8. Assign users/groups who should have access

Critical: Th Entity ID in Azur must **exactly match** `SAML_SP_ENTITY_ID` in your `.env` fil .

2.6 Step 6: Configure Traefik

Ensur `docker-compose.y ml` has prop r Tra fik configuration:

```
# Key Traefik labels for hydra-saml-auth service:
labels:
  - "traefik.enable=true"
  - "traefik.http.routers.hydra.rule=Host(‘hydra.yourdomain.edu‘)"
  - "traefik.http.routers.hydra.entrypoints=websecure"
  - "traefik.http.routers.hydra.tls=true"
  - "traefik.http.services.hydra.loadbalancer.server.port=6969"
```

2.7 Step 7: Start Services

```
# Build and start all services
docker compose build
docker compose up -d

# Verify services are running
docker compose ps

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

2.8 Step 8: Verify Installation

```
# Test HTTPS access
curl -I https://hydra.yourdomain.edu/

# Should redirect to Azure AD login (302 to login.microsoftonline.com)

# Check JWKS endpoint
curl https://hydra.yourdomain.edu/.well-known/jwks.json
```

3 Post-Installation Setup

3.1 Create Admin User

After first login via SAML, promote a user to admin:

```
# Access database
sqlite3 /app/data/webui.db

# Find user ID
SELECT id, email, role FROM users WHERE email LIKE '%admin%';

# Set as admin
UPDATE users SET role = 'admin' WHERE email = 'admin@yourdomain.edu';
```

3.2 Test Container Creation

1. Log in to https://hydra.yourdomain.edu/dashboard
2. Navigate to "Containers" tab
3. Click "Initialize Container"
4. Verify VS Code and Jupyter are accessible

3.3 Configure Backup Cron

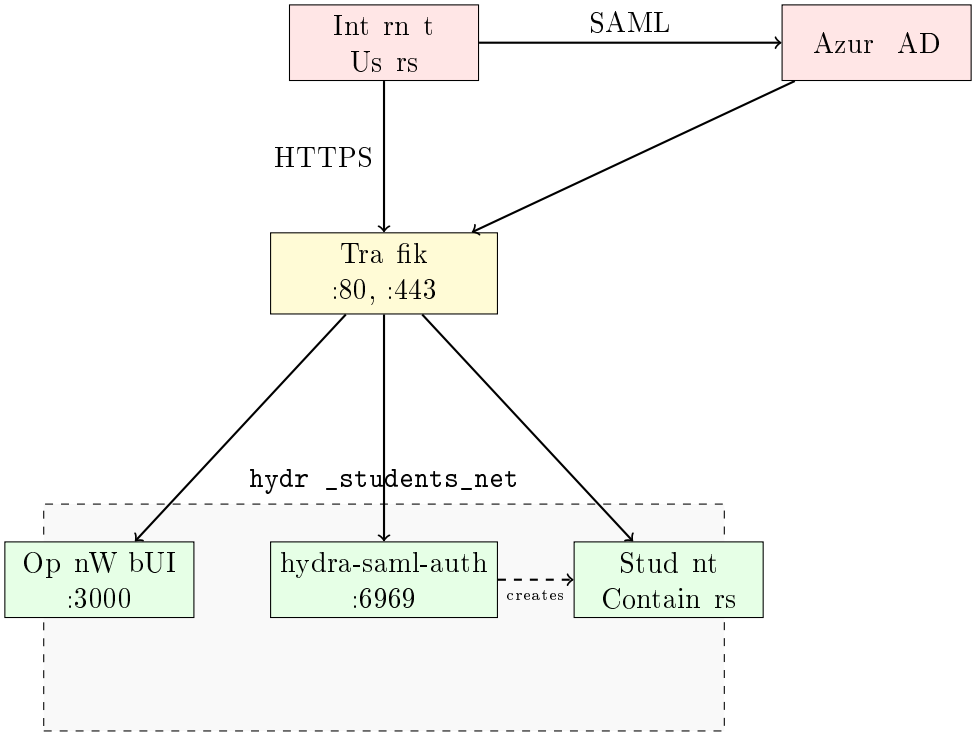
```
# Add to root crontab
crontab -e

# Daily database backup at 2 AM
0 2 * * * docker exec hydra-saml-auth sqlite3 /app/data/webui.db ".
    backup '/backups/hydra-$(date +\%Y\%m\%d).db'"

# Weekly cleanup of old backups
0 3 * * 0 find /backups -name "hydra-*.db" -mtime +30 -delete
```

4 Network Architecture

4.1 Deployment Diagram



4.2 Port Mapping

Service	Internal	External	Protocol
Tra fik	-	80, 443	HTTP/HTTPS
hydra-saml-auth	6969	via Tra fik	HTTP
Op nW bUI	3000	via Tra fik	HTTP
Stud nt VS Cod	8443	/stud nts/{us r}/vscod	HTTP
Stud nt Jupyt r	8888	/stud nts/{us r}/jupyt r	HTTP

5 Security Configuration

5.1 TLS Certificates

For L t's Encrypt (r comm nd d):

```
# Traefik handles automatic certificate management
# Add to docker-compose.yaml traefik command:
command:
  - "--certificatesresolvers.letsencrypt.acme.email=admin@yourdomain.
      edu"
  - "--certificatesresolvers.letsencrypt.acme.storage=/letsencrypt/acme
      .json"
  - "--certificatesresolvers.letsencrypt.acme.httpchallenge.entrypoint=
      web"
```

For institutional c rtificat s:

```
# Mount certificates in Traefik
```

```
volumes:
  - ./certs/cert.pem:/certs/cert.pem:ro
  - ./certs/key.pem:/certs/key.pem:ro

# Configure in dynamic config
tls:
  certificates:
    - certFile: /certs/cert.pem
      keyFile: /certs/key.pem
```

5.2 Container Security

Stud nt contain rs run in **privileged mode** by d fault to nabl Dock r-in-Dock r. This grants significant acc ss. Consid r:

- R stricting to trust d us rs only
- Monitoring contain r activity
- Impl m nting r sourc quotas

6 Useful Aliases

Add to `~/.b shrc`:

```
# ===== HYDRA MANAGEMENT ALIASES =====

# Docker shortcuts
alias dps='docker ps --format "table {{.Names}}\t{{.Status}}\t{{.Ports
    }}"'
alias dlogs='docker compose logs -f'
alias dexec='docker exec -it'

# Hydra-specific
alias hydra-logs='docker compose logs -f hydra-saml-auth'
alias hydra-restart='docker compose restart hydra-saml-auth'
alias hydra-rebuild='docker compose build hydra-saml-auth && docker
    compose up -d hydra-saml-auth'

# Student container management
alias students='docker ps --filter "name=student-" --format "table {{.
    Names}}\t{{.Status}}"'
alias student-logs='docker logs -f'
alias student-shell='docker exec -it'
alias student-stop='docker stop'
alias student-rm='docker rm -f'

# Find student by partial name
findstudent() {
    docker ps --filter "name=student-" --format "{{.Names}}" | grep -i
        "$1"
}

# Get student container stats
student-stats() {
    docker stats --no-stream --filter "name=student-"
```

```
}

# Backup database
backup-db() {
    docker exec hydra-saml-auth sqlite3 /app/data/webui.db ".backup '/
        backups/hydra-$(date +%Y%m%d-%H%M%S).db'"
    echo "Backup created: hydra-$(date +%Y%m%d-%H%M%S).db"
}

# Quick health check
hydra-health() {
    echo "=== Services ==="
    docker compose ps
    echo ""
    echo "=== Student Containers ==="
    docker ps --filter "name=student-" --format "table {{.Names}}\t{{.
        Status}}\t{{.RunningFor}}"
    echo ""
    echo "=== Resource Usage ==="
    docker stats --no-stream --format "table {{.Name}}\t{{.CPUPerc}}\t
        {{.MemUsage}}" | head -10
}
```

Apply chang s:

```
source ~/.bashrc
```

7 Monitoring

7.1 Log Locations

Component	Command
Main s rvic	`docker compose logs hydr -s ml- uth`
Tra fik	`docker compose logs tr efik`
Stud nt contain r	`docker logs student-<usern me>`
All s rvic s	`docker compose logs -f`

7.2 Health Endpoints

```
# Check main service
curl https://hydra.yourdomain.edu/health

# Check JWKS (JWT verification)
curl https://hydra.yourdomain.edu/.well-known/jwks.json

# Check Traefik dashboard (if enabled)
curl http://localhost:8080/api/overview
```

8 Troubleshooting

8.1 Common Issues

Issue	Solution
SAML login fails	V rify `METADATA_URL` acc ssibl , Entity ID match s xactly
Stud nt contain r 404	Ch ck contain r on `hydr _students_net`, Tra - fik running
Contain r won't start	V rify `hydr -student-cont iner:l test` im- ag built
P rmission d ni d	Ch ck Dock r sock t p rmissions, us r in dock r group
Databas lock d	R start s rvic , ch ck for multipl writ rs

8.2 Debug Commands

```
# Check Docker networks
docker network ls
docker network inspect hydra_students_net

# Verify image exists
docker images | grep hydra-student-container

# Check container networking
docker exec student-<user> curl -I http://hydra-saml-auth:6969/

# View Traefik routes
docker exec traefik cat /etc/traefik/traefik.yml
```

9 Upgrade Procedures

9.1 Updating the Application

```
# Pull latest code
git pull origin main

# Rebuild and restart
docker compose build hydra-saml-auth
docker compose up -d hydra-saml-auth

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

9.2 Updating Student Container Image

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

Existing stud nt contain rs continu using th old imag . Stud nts must d l t and r cr at th ir contain r to g t th updat d imag .

10 References

- Dock r Docum ntation: https://docs.docker.com/
- Dock r Compos : https://docs.docker.com/compose/
- Tra fik Docum ntation: https://doc.tr efik.io/tr efik/
- Azur AD SAML S tup: https://docs.microsoft.com/en-us/ zure/ ctive-directory/develop/single-sign-on-s ml-protocol
- passport-saml: https://github.com/node-s ml/p ssport-s ml
- cod -s rv r: https://coder.com/docs/code-server/l test
- Jupyt r: https://jupyter.org/document tion
- L t's Encrypt: https://letsencrypt.org/docs/

Installation Complete!

Aft r compl ting th s st ps:

1. Visit `https://hydr .yourdom in.edu/login`
2. Auth nticat via Azur AD
3. Navigat to th dashboard
4. Cr at a t st contain r to v rify th s tup

For ongoing manag m nt, r f r to th **Hydra Infrastructure Management Guide**.