HDFS Cluster Deployment

Production Cluster Architecture & Hands-On Access

Professor Anis Koubaa

SE 446 - Big Data Engineering
Prince Sultan University

Spring 2026

Outline

1. Cluster Overview
2. Cluster Architecture
3. Security Features
4. Web UI Overview
5. Hands-On Learning
6. Monitoring
7. Best Practices
8. Course Integration
9. Summary

Your Production HDFS Cluster

Welcome to Real Big Data Infrastructure!

You now have access to a **production-grade HDFS cluster** deployed on DigitalOcean.

Cluster Specifications:

- **3 Nodes**: 1 Master + 2 Workers
- **Hadoop**: Version 3.4.1
- **Storage**: 95.66 GB total
- **Replication**: Factor 2
- **Security**: SSL + Authentication

Access Information:

- **URL**: hdfs.aniskoubaa.org
- **Username**: xxxxxxx
- **Password**: xxxxxxx
- **Protocol**: HTTPS (Secure)

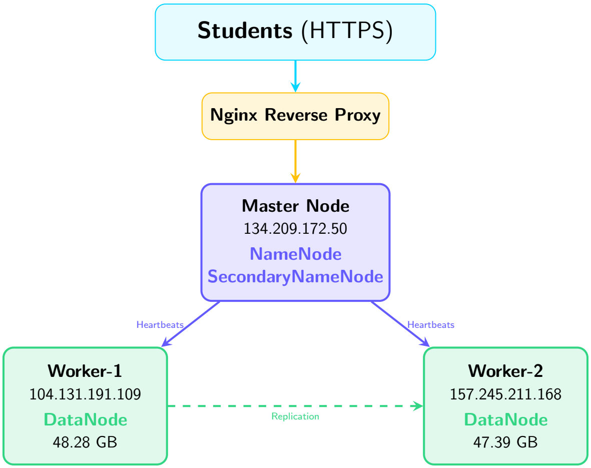
Live Cluster

This is a **real distributed system** — not a simulation!

Why a Real Cluster?

1. **Authentic Experience**
   Work with the same tools used in industry
2. **Understand Distribution**
   See data physically split across multiple machines
3. **Observe Replication**
   Watch blocks replicate to different nodes
4. **Monitor Real Metrics**
   Track storage, network, and health in real-time
5. **Hands-On Learning**
   Upload files, run commands, explore the Web UI

Cluster Topology



Students (HTTPS)

Nginx Reverse Proxy

Master Node
134.209.172.50
NameNode
SecondaryNameNode

Worker-1
104.131.191.109
DataNode
48.28 GB

Worker-2
157.245.211.168
DataNode
47.39 GB

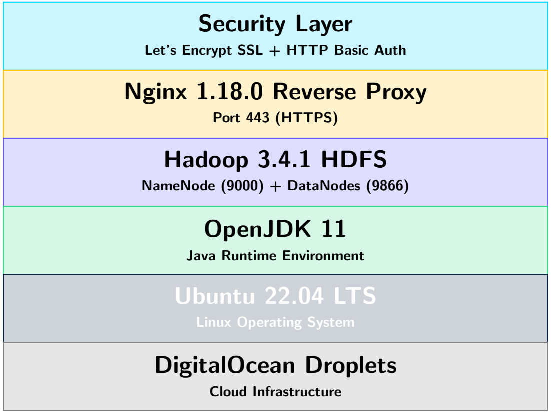
Node Details

Node	IP Address	Role	Components
master-node	134.209.172.50	Master	NameNode, SecondaryNameNode
worker-node-1	104.131.191.109	Worker	DataNode (48.28 GB)
worker-node-2	157.245.211.168	Worker	DataNode (47.39 GB)
Total	-	-	**95.66 GB**

Replication Factor: 2

Each block is stored on **2 different DataNodes** for fault tolerance.
Effective capacity: **47.83 GB** (half of total due to replication)

Software Stack



Security Layer

Let's Encrypt SSL + HTTP Basic Auth

Nginx 1.18.0 Reverse Proxy

Port 443 (HTTPS)

Hadoop 3.4.1 HDFS

NameNode (9000) + DataNodes (9866)

OpenJDK 11

Java Runtime Environment

Ubuntu 22.04 LTS

Linux Operating System

DigitalOcean Droplets

Cloud Infrastructure

Multi-Layer Security

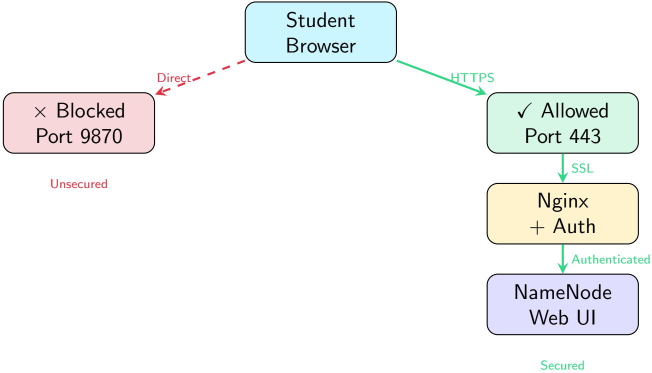
Why Security Matters

Production clusters must protect data from unauthorized access.

Security Layers Implemented:

1. **SSL/TLS Encryption**
   All communication encrypted using HTTPS
   ✓ Certificate: Let's Encrypt (Valid until Apr 27, 2026)
2. **HTTP Basic Authentication**
   Username/password required for Web UI access
   ✓ Credentials: xxxxx / xxxxxx
3. **Firewall Protection**
   Direct HDFS ports blocked from internet
   ✓ Only HTTPS (443) accessible publicly
4. **Worker Isolation**
   DataNode ports restricted to master node only

Access Flow



Result

Only secure, authenticated access allowed — just like enterprise systems!

Accessing the Web UI

Step-by-Step Access

1. Open browser: `https://hdfs.aniskoubaa.org`
2. Enter credentials when prompted:
   - Username: `xxxxxxx`
   - Password: `xxxxxxx`
3. Click *Sign In*

Browser Warning

If you see "Your connection is not private," this is normal for educational SSL certificates. Click *Advanced* → *Proceed to hdfs.aniskoubaa.org*

Try it now! Open the cluster and explore.

Web UI Overview Page

What You'll See:

Cluster Summary:

- Configured Capacity
- DFS Used / Remaining
- Live / Dead Nodes
- Number of blocks
- Missing blocks (should be 0)

NameNode Information:

- Started time
- Version: 3.4.1
- Compiled date
- Cluster ID

Key Metrics to Watch:

- **Live Nodes: 2**
  Both DataNodes healthy
- **Dead Nodes: 0**
  No failures
- **DFS Used**
  Storage consumed
- **Blocks**
  Total data blocks

Exercise

Datanodes Tab

Navigation: Click **Datanodes** tab

Information Displayed:

Column	Description
Node	IP address and hostname of DataNode
Last Contact	Time since last heartbeat (should be seconds)
Admin State	In Service / Decommissioned
Capacity	Total storage available on this node
Used	Storage consumed by HDFS blocks
Non DFS Used	Storage used by other files
Remaining	Available storage
Blocks	Number of blocks stored on this node

Observe

Notice blocks are **distributed** across both DataNodes!

Utilities: Browse the File System

Navigation: *Utilities* → **Browse the file system**

What You Can Do:

- Browse HDFS directories (like file explorer)
- View file metadata:
  - File size
  - Replication factor
  - Block size
  - Owner and permissions
- See which **DataNodes** store each block
- Download files
- View file contents (for text files)

Try It

1. Navigate to / (root directory)
2. Look for any files or directories

Block Information

For any file, you can see:

File Properties:

- File path
- Size (bytes)
- Block size (default 128 MB)
- Replication (2 in our cluster)
- Number of blocks

Block Locations:

- Block ID
- DataNode addresses
- Storage type
- Block pool ID

Example: 200 MB file

- Split into: 2 blocks (128 MB + 72 MB)
- Each block replicated to: 2 DataNodes
- Total blocks in cluster: 4 (2 blocks × 2 replicas)
- Total storage used: 400 MB (200 MB × 2 replicas)

Connecting via SSH (Optional)

For Advanced Users: Access cluster via command line

SSH to Master Node

```
ssh root@134.209.172.50
# Password: [Contact instructor]
```

Switch to Hadoop user:

```
sudo su - hadoop
```

Check cluster status:

```
hdfs dfsadmin -report
```

Note

SSH access is **optional**. Most exercises use the Web UI.
Command-line access provided for those interested in deeper exploration.

HDFS Command Examples

If connected via SSH:

```
# List files in HDFS root
hdfs dfs -ls /

# Create a directory
hdfs dfs -mkdir /student_data

# Upload a file from local to HDFS
hdfs dfs -put myfile.txt /student_data/

# View file content
hdfs dfs -cat /student_data/myfile.txt

# Check file status
hdfs dfs -stat "%r␣%b␣%n" /student_data/myfile.txt
# Output: replication, block_size, filename

# Download file from HDFS
hdfs dfs -get /student_data/myfile.txt ./local_copy.txt
```

Observe in Web UI

After running commands, refresh the Web UI to see changes!

Learning Exercises

Use the cluster to verify HDFS concepts:

1. **Block Distribution**
   Upload a 200 MB file → Check Web UI → See blocks on different DataNodes
2. **Replication Factor**
   View any file → Count replicas → Verify replication $= 2$
3. **Storage Calculation**
   Upload 100 MB file → Check DFS Used → Verify 200 MB used (100 MB × 2)
4. **Heartbeats**
   Datanodes tab → Check "Last Contact" → Should be < 10 seconds
5. **Capacity Monitoring**
   Overview tab → Watch DFS Used vs Remaining as you add files

Experiment Ideas

Advanced Experiments (Optional):

1. Compare File Formats

- Upload same data as CSV and Parquet
- Compare file sizes
- Observe compression benefits

2. Block Size Impact

- Upload small file ($<$ 128 MB) → How many blocks?
- Upload large file ($>$ 128 MB) → How is it split?
- Calculate: $\text{blocks\_needed} = \lceil \text{filesize} \; / \; \text{blocksize} \rceil$

3. Fault Tolerance Simulation

- Note which DataNodes store a block
- Imagine DataNode-1 fails
- Data still available on DataNode-2 (that's replication!)

Key Metrics to Monitor

Metric	Healthy Value	What It Means
Live Nodes	2	Both DataNodes operational
Dead Nodes	0	No failures detected
DFS Used %	$< 80\%$	Adequate free space
Under-replicated blocks	0	All blocks properly replicated
Missing blocks	0	No data loss
Corrupt blocks	0	Data integrity maintained
Last Contact	< 10 sec	Heartbeats arriving

Warning Signs

- Dead Nodes $> 0 \rightarrow$ DataNode failure
- Missing blocks $> 0 \rightarrow$ Potential data loss
- DFS Used $> 90\% \rightarrow$ Running out of space

Common Issues & Solutions

Issue: Cannot Access Web UI

Solutions:

- Check URL: `https://` (not http)
- Verify credentials: xxxxxx / xxxxxxx
- Try different browser
- Accept SSL certificate warning

Issue: "Safe Mode" Message

Meaning: NameNode is in read-only mode (startup or maintenance)
Solution: Wait a few minutes; system will auto-exit safe mode

Issue: Slow Performance

Possible Causes:

- Network congestion (multiple students uploading)

Cluster Usage Guidelines

Do's ✓

- Upload files for learning purposes
- Experiment with different file sizes
- Share the cluster respectfully
- Monitor your storage usage
- Clean up test files when done

Don'ts ×

- Upload sensitive/copyrighted content
- Delete other students' files
- Bypass security
- Upload files > 1 GB

File Organization

Suggested Directory Structure:

```
/
+-- /student_<yourname>/ # Your personal folder
| +-- /test_data/ # Test files
| +-- /lab_assignments/ # Lab work
| +-- /experiments/ # Your experiments
+-- /shared/ # Class shared folder
| +-- /datasets/ # Common datasets
+-- /tmp/ # Temporary files
```

Example: Create Your Folder

```
hdfs dfs -mkdir /student_ahmed
hdfs dfs -mkdir /student_ahmed/test_data
```

Organized structure helps everyone find their work!

Data Management

Storage Best Practices:

1. **Check capacity before uploading**

```
hdfs dfs -df -h /
```

2. **Remove files you no longer need**

```
hdfs dfs -rm /student_yourname/old_file.txt
```

3. **Use descriptive filenames**

   Good: `lab2_temperature_data.csv`

   Bad: `data.csv`

4. **Remember replication multiplier**

   Uploading 500 MB uses 1 GB (500 MB × 2 replicas)

How This Cluster Supports Your Learning

Theoretical Concepts:

- NameNode metadata
- DataNode storage
- Block distribution
- Replication factor
- Heartbeat mechanism
- Rack awareness
- Data integrity

Practical Verification:

- See metadata in Web UI
- Watch blocks spread across nodes
- Calculate storage with replication
- Monitor heartbeats (Last Contact)
- Verify block checksums
- Observe live/dead nodes
- Test fault tolerance

Learn by Doing

Every concept from lectures can be **verified on this cluster!**

Lab Assignments

Upcoming labs will use this cluster:

1. **Lab 1: HDFS Basics**
   - Upload files, observe block distribution
   - Calculate storage with replication
   - Explore Web UI features
2. **Lab 2: File Formats**
   - Compare CSV vs Parquet
   - Measure compression ratios
   - Analyze query performance
3. **Lab 3: MapReduce (Later)**
   - Run MapReduce jobs on cluster data
   - Process distributed datasets
   - Monitor job execution

Resources

Documentation & Help:

- **Cluster Access**: https://hdfs.aniskoubaa.org
- **Administrator Guide**: See course repository `cluster_setup/latex/hdfs_admin_guide.pdf`
- **Quick Reference**: `cluster_setup/QUICK_REFERENCE.md`
- **Hadoop Documentation**: hadoop.apache.org/docs/r3.4.1/
- **Office Hours**: For cluster issues or questions
- **Discussion Forum**: Share tips with classmates

Getting Help

Problem with cluster? Email instructor with:

- What you were trying to do
- Error message (screenshot)
- Time of occurrence

Key Takeaways

1. **Real Cluster Access**
   You have a production HDFS cluster at hdfs.aniskoubaa.org
2. **Architecture**
   3 nodes: 1 NameNode + 2 DataNodes, 95.66 GB total, replication factor 2
3. **Security**
   SSL encryption + authentication + firewall protection
4. **Web UI**
   Monitor metrics, browse files, view block locations
5. **Hands-On Learning**
   Upload files, run commands, verify theoretical concepts
6. **Best Practices**
   Organize files, respect shared resources, monitor usage

Next Steps

Immediate Actions

1. Access cluster: https://hdfs.aniskoubaa.org
2. Log in: xxxxxxx / xxxxxxxx
3. Explore the Web UI
4. Check Overview, Datanodes, and Utilities tabs
5. Browse the file system

This Week

- Practice HDFS commands (if using SSH)
- Create your personal directory
- Upload a test file
- Observe block distribution
- Calculate storage with replication

Quick Access Reference

Bookmark This!

URL:	`https://hdfs.aniskoubaa.org`
Username:	`xxxxxxxx`
Password:	`xxxxxxxxx`
SSH (Optional):	`ssh root@134.209.172.50`

Remember

Secure • Shared • Educational

Questions?

Prof. Anis Koubaa
akoubaa@psu.edu.sa

Now let's log in and explore the cluster together!