

Lab 2.3: Backup and Restore

Objective:

In this lab, you will **practice taking a complete backup of an XWiki instance**, including the application data and the database. You will also simulate a **restore operation** in a different environment to ensure your backup is functional.

By the end of this lab, you will:

- Understand which components of XWiki need to be backed up
- Create a manual backup of XWiki's file system and database
- Perform a full restore on another instance

Step-by-Step Instructions

1. Understand What Needs to Be Backed Up

 An XWiki installation consists of:

- **XWiki home directory** (Tomcat `webapps/xwiki`, configuration, attachments, extensions)
- **XWiki database** (MySQL/PostgreSQL): stores pages, user data, settings, and history

To fully back up XWiki, **you must back up both the file system and the database**.

2. Backup Process

2.1. Backup XWiki Home (Files)

 Target: WAR-deployed `xwiki/` directory and configuration files.

```
bash

# Stop Tomcat service before backup
sudo systemctl stop tomcat

# Create a compressed backup of the XWiki directory
tar -czvf xwiki-home-backup.tar.gz /path/to/tomcat/webapps/xwiki

# Optionally, backup other important config directories:
tar -czvf tomcat-conf-backup.tar.gz /path/to/tomcat/conf
```

2.2. Backup the Database

For MySQL:

```
bash
```

```
mysqldump -u xwiki -p xwiki > xwiki-db-backup.sql
```

For PostgreSQL:

```
bash
```

```
pg_dump -U xwiki xwiki > xwiki-db-backup.sql
```

 Tip: Ensure you back up with a user that has full DB privileges.

3. Restore Process on a Different Setup



This step simulates a disaster recovery or migration scenario.

3.1. Prepare New Environment

- Install Java, Tomcat, and MySQL/PostgreSQL (see **Lab 2.1**)
- Deploy the same **XWiki WAR** version

3.2. Restore XWiki Home

```
bash
```

```
# Extract backup into new Tomcat webapps directory
tar -xzvf xwiki-home-backup.tar.gz -C /path/to/new-tomcat/webapps/
```

 Ensure the `xwiki/` directory is placed directly in `webapps/`.

3.3. Restore the Database

For MySQL:

```
bash
```

```
mysql -u root -p
CREATE DATABASE xwiki CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
CREATE USER 'xwiki'@'localhost' IDENTIFIED BY 'xwiki_pass';
GRANT ALL PRIVILEGES ON xwiki.* TO 'xwiki'@'localhost';
EXIT;
```

```
mysql -u xwiki -p xwiki < xwiki-db-backup.sql
```

For PostgreSQL:

```
bash
```

```
psql -U postgres
CREATE DATABASE xwiki;
CREATE USER xwiki WITH ENCRYPTED PASSWORD 'xwiki_pass';
GRANT ALL PRIVILEGES ON DATABASE xwiki TO xwiki;
\q
```

```
psql -U xwiki xwiki < xwiki-db-backup.sql
```

3.4. Launch the Restored XWiki Instance

1. Start Tomcat:

```
bash  
sudo systemctl start tomcat
```

2. Open your browser:

👉 <http://localhost:8080/xwiki>

3. Verify:

- Admin login works
- Pages, attachments, and settings are restored
- Installed extensions are present

Lab Summary

In this lab, you learned how to:

- Identify which components of XWiki need to be backed up
- Create and restore a backup of the XWiki file system
- Perform a complete database dump and restore operation
- Launch a restored instance successfully

Optional Advanced Tasks

- **Automate backup with a script** and cron job
- Use `rsync` for incremental backups
- Restore to a **different database engine** (e.g., migrate from MySQL to PostgreSQL)
- Perform a test restore **without overwriting** your live instance (e.g., use Docker or local VM)

Helpful Tips

Task	Command / Notes
Check available DBs	<code>mysql -e "SHOW DATABASES;" or psql -l</code>
Verify DB size	<code>du -sh /var/lib/mysql/xwiki</code>
Compress large SQL files	<code>gzip xwiki-db-backup.sql</code>
Schedule backup (Linux)	Use <code>crontab -e</code>