CSC-440 Secure Software Engineering

Nextcloud Security Lab

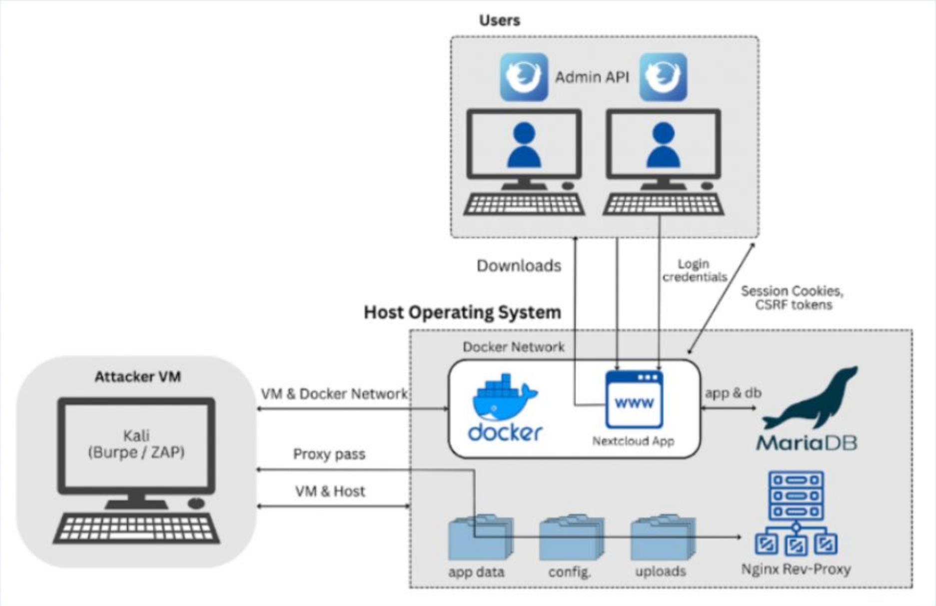
Team 7
Jason Sonith
Izabel Valdez
GiaGia Diep
Mina Dang

Final Project

FALL 2025

Project Overview & Goals

Nextcloud is a vulnerable open source platform for secure file storage, syncing, and collaboration, similar to Google Drive but fully self hosted. In this project, Nextcloud is deployed using Docker, which runs the app, database, and proxy in isolated containers to mimic a real production environment. The purpose of the project is to evaluate the security posture of this cloud storage system by examining risks to user data, authentication, file handling, network traffic, and container isolation, and to determine how safe the deployment is for real world use.



Hands-On Deployment & Testing

ANALYZE SECURITY OF A CLOUD-STORAGE PLATFORM USING HANDS-ON TESTING

Threat Modeling & Assessment

FIND VULNERABILITIES IN ARCHITECTURE, SESSIONS, AUTHENTICATION, AND KEYS

Hardening & Verification

APPLY MITIGATIONS AND EVALUATE IMPROVEMENT

ZAP by Checkmarx **ZAP Scanning Report**

Alerts

Name	Risk Level	Number of Instances
Vulnerable JS Library	High	1
CSP: Failure to Define Directive with No Fallback	Medium	2
CSP: Wildcard Directive	Medium	2
CSP: style-src unsafe-inline	Medium	5
Deprecated Feature Policy Header Set	Low	3
Insufficient Site Isolation Against Spectre Vulnerability	Low	12
Permissions Policy Header Not Set	Low	5
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	Low	11
Server Leaks Version Information via "Server" HTTP Response Header Field	Low	11
Timestamp Disclosure - Unix	Low	21
Information Disclosure - Suspicious Comments	Informational	3

Methodology

WEEK 1 : ENVIRONMENT SETUP

- Built the initial 3 container lab (Nextcloud, MariaDB, nginx) and confirmed services were running.
- Verified container networking, login capability, and baseline architecture.

WEEK 2: ACCOUNTS, PERMISSIONS & SURFACE MAPPING

- Created lab users and configured group roles with least privilege access.
- Performed first Nmap scan where only port 8080 is exposed.
- Validated dashboard reachability and user authentication.

WEEK 3 : CONFIGURATION, SECRETS, AND HTTP SECURITY REVIEW

- Analyzed config.php and identified plaintext secrets (passwordsalt, secret, DB password).
- Reviewed .env files and environment variable exposure risks.
- Noted lack of HTTPS, secure cookies, HSTS, and encryption settings.

WEEK 4 : AUTHENTICATION, SESSION TESTING & ZAP SCAN

- Performed authentication and session handling tests with Burp Suite (cookies, CSRF, password policy, brute-force behavior).
- Conducted OWASP ZAP Baseline Scan and crawled 32 URLs, passed 56/67 rules, 0 high-risk findings, 11 low risk warnings
- Documented session cookie behavior and security flag configurations.

Methodology

```yaml
services:
  # ============================================================================
  # DATABASE: MariaDB 11.8.5
  # ============================================================================
  # Role: Stores Nextcloud user data, files metadata, and configuration
  # Security: Internal only (not exposed to host network)
  # ----------------------------------------------------------------------------
  db:
    image: mariadb:11.8.5
    # ^ PINNED VERSION: MariaDB 11.8.5 LTS (released 2024-11-14)
    # ^ Previous: mariadb:11 (floating tag - INSECURE)
    # ^ CVEs remaining: 27 (0 Critical, 4 High in gosu binary - low risk)

    restart: always

    command: --transaction-isolation=READ-COMMITTED --binlog-format=ROW
    # ^ Required by Nextcloud for proper transaction handling

 # ============================================================================
 # APPLICATION: Nextcloud 29-apache
 # ============================================================================
 # Role: Main Nextcloud application (PHP + Apache)
 # Security: Exposed on port 8080 (HTTP), proxied via nginx on 443 (HTTPS)
 # ----------------------------------------------------------------------------
 app:
   image: nextcloud:29-apache
   # ^ Nextcloud 29.0.16.1 with PHP 8.2.29
   # ^ Includes patches for CVE-2024-2756 (PHP heap overflow)
   # ^ Note: Using floating tag to maintain compatibility with existing data

   restart: always

   ports:
     - "0.0.0.0:8080:80"
     # ^ Direct HTTP access (for testing/debugging)
     # ^ Production: Remove this and use only nginx proxy on 443
```

WEEK 5 : CONTAINER SECURITY & HARDENING BASELINE

- Performed comprehensive file-handling security tests including malicious file uploads, double extensions, large-file DoS attempts, and WebDAV file operations.
- Tested filename and content robustness using Unicode names, long filenames, HTML/JS payloads, SVG scripts, and malware test files
- Reviewed container security posture through Docker inspection, CIS Docker Benchmark checks, privilege escalation risks, and secret management review.

WEEK 6 : FULL CVE ASSESSMENT (TRIVY + MANUAL REVIEW)

- Ran Trivy on all containers and identified 187+ unique CVEs, including 21 critical.
- Highlighted high-risk CVEs
- Prioritized remediation based on severity and exploitability.

WEEK 7 : HARDENING, PATCHING & REBUILDS

- Patched all Priority 1 CVEs by updating Nextcloud, MariaDB, and nginx images to their fixed versions and rebuilding the entire environment.
- Applied full container hardening including dropping all capabilities, enforcing no-new-privileges, enabling read-only filesystems, running containers as non-root, tightening security options, and adding CPU/memory limits.
- Re-scanned and validated fixes with fresh Trivy scans, compared before and after CVE counts, documented remediation evidence, created the hardened docker-compose.yml, and prepared final report deliverables.

Vulnerabilities & CVEs

These five vulnerabilities represent the highest-risk weaknesses discovered because they allow remote compromise, privilege escalation, or full system takeover across core components used by the entire Nextcloud stack.

1. **CVE-2024-3094: xz-utils Supply-Chain Backdoor (CVSS 10.0)**
   - A malicious backdoor in allowed remote SSH compromise with no authentication, enabling full system takeover.
2. **CVE-2023-3446: OpenSSL Side-Channel Attack (CVSS 9.8)**
   - A remote attacker could recover TLS private keys, enabling man-in-the-middle interception of encrypted Nextcloud traffic.
3. **CVE-2022-37454: zlib Buffer Overflow (CVSS 9.8)**
   - Malformed compressed data could corrupt memory and lead to remote code execution in services using zlib (PHP, curl, SSH)
4. **CVE-2023-4911: glibc "Looney Tunables" Privilege Escalation (CVSS 9.8)**
   - A crafted environment variable allowed local privilege escalation, enabling a compromised container to escape to the host.
5. **CVE-2024-2756: PHP Heap Buffer Overflow (CVSS 9.8)**
   - A memory-corruption flaw in PHP 8.x could allow remote code execution via crafted payloads processed by the PHP runtime.

CVE ID	Component	CVSS	Description	Status
CVE-2024-3094	xz-utils	10.0	Backdoor allowing remote compromise	Remediated
CVE-2023-3446	OpenSSL	9.8	Cryptographic key recovery	Remediated
CVE-2022-37454	zlib	9.8	Buffer overflow in compression	Remediated
CVE-2023-4911	glibc	7.8	Privilege escalation to root	Remediated
CVE-2024-2756	PHP	7.5	Heap overflow in PHP runtime	Remediated

Results

1) Eliminated critical security vulnerabilities
Critical CVEs to 0 after patching and image updates.

2) Reduced vulnerability count by more than 70%
From over 2,000+ CVEs down to approximately 50 low risk items.

3) Strengthened the container security across the stack
Linux capabilities reduced by around 70%, added CPU and RAM limits, enforced no new privileges, and applied read only filesystem to the proxy.

4) Improved application-level security significantly
Verified strong password policy, effective brute force protection, secure session cookies, CSRF validation, and XSS protections.

5) Hardened Docker environment to meet CIS Benchmark controls
Passed user namespace, capability restrictions, resource limits, and privilege escalation protections.

6) Overall system risk reduced from CRITICAL to LOW
Deployment after Week 6 is suitable for real world use with ongoing maintenance.

```
CONTAINER HARDENING VERIFICATION
Generated: 2025-11-25 17:05:49
==================================

1. SECURITY OPTIONS
-------------------
Proxy (nginx):
  - Security Options: [no-new-privileges:true]
  - Read-Only Filesystem: true
  - Capabilities Dropped: [ALL]
  - Capabilities Added: [CAP_CHOWN CAP_NET_BIND_SERVICE
  - User: 101:101

App (Nextcloud):
  - Security Options: [no-new-privileges:true]
  - Capabilities Dropped: [ALL]
  - Capabilities Added: [CAP_CHOWN CAP_DAC_OVERRIDE CAP_

DB (MariaDB):
  - Security Options: [no-new-privileges:true]
  - Capabilities Dropped: [ALL]
  - Capabilities Added: [CAP_CHOWN CAP_DAC_OVERRIDE CAP_

2. RESOURCE LIMITS
------------------
6794d1cb54ef - CPU: 0.00% - Memory: 12.85MiB / 512MiB
0424e7bfcddb - CPU: 0.00% - Memory: 92.04MiB / 2GiB
7d3fe34350c5 - CPU: 0.01% - Memory: 103.4MiB / 2GiB
```

Conclusion

1

The security posture of the Docker-deployed Nextcloud environment improved dramatically

Hardening efforts reduced critical risks, strengthened container isolation, and removed exploitable vulnerabilities.

2

After remediation, the system meets the requirements for a more secure, self hosted cloud platform

All critical vulnerabilities were remediated, CIS Docker controls were satisfied, and key application protections were validated.

3

The final deployment is stable, resilient, and suitable for real-world use with ongoing monitoring

Remaining issues are low risk, and continued scanning, updates, and certificate management will maintain security over time.