

ORGANIZATION OF USER ACCESS RIGHTS MANAGEMENT SYSTEMS IN EDUCATIONAL INFORMATION SYSTEMS

Vinnitsia National Technical University

Abstract

The research paper examines theoretical and practical aspects of organizing user access rights management systems in educational information systems. The main access control models (DAC, MAC, RBAC, ABAC) are analyzed, and their advantages and limitations in the context of educational platforms are identified. Based on the developed requirements for a student learning materials exchange platform, the specifics of the role model for three user classes are studied: student, instructor, and administrator. The selection of a hybrid RBAC+ABAC model as optimal for systems with dynamic content and social interaction is justified. Key security threats are identified and technical solutions are proposed to address them.

Keywords: access control, RBAC, ABAC, educational information systems, privilege separation, data security, user roles.

Анотація

У тезах розглянуто теоретичні та практичні аспекти організації системи розмежування прав доступу користувачів в освітніх інформаційних системах. Проаналізовано основні моделі контролю доступу (DAC, MAC, RBAC, ABAC), визначено їх переваги та обмеження в контексті освітніх платформ. На основі розроблених вимог до платформи обміну навчальними матеріалами між студентами досліджено специфіку рольової моделі для трьох класів користувачів: студент, викладач та адміністратор. Обґрунтовано вибір гібридної RBAC+ABAC моделі як оптимальної для систем із динамічним контентом та соціальною взаємодією. Визначено ключові загрози безпеки та запропоновано технічні рішення для їх усунення.

Ключові слова: контроль доступу, RBAC, ABAC, освітні інформаційні системи, розмежування прав, безпека даних, ролі користувачів.

Introduction

The rapid digitalization of higher education has led to widespread adoption of educational information systems (EIS) – platforms that manage learning materials, facilitate student-instructor interaction, and store sensitive academic data. With increasing numbers of users and growing content repositories, ensuring proper access control has become a critical engineering challenge [1].

Access rights management in educational contexts differs fundamentally from corporate environments. Educational platforms must simultaneously support open knowledge-sharing culture while enforcing strict boundaries – a student must freely browse and upload materials but cannot delete another user's content; an instructor must publish official course materials but cannot access administrative reports; an administrator must moderate all content but must be prevented from impersonating other users. These requirements demand a carefully designed, multi-layered access control architecture [2].

The relevance of this topic is confirmed by statistics: according to a 2023 report by EDUCAUSE, 67% of higher education institutions reported unauthorized data access incidents related to insufficient privilege separation in their digital learning environments [3]. Furthermore, the adoption of peer-to-peer learning platforms – where students themselves produce and share academic content – introduces novel access control challenges not addressed by traditional LMS (Learning Management System) security models. This research paper investigates access control models applicable to educational platforms, with a particular focus on a student learning materials exchange system developed through systematic requirements analysis.

Access control models

Access control models define rules that govern how subjects (users) interact with objects (resources). Four principal models are relevant to educational information systems [4].

Discretionary Access Control (DAC) grants resource owners the right to define access permissions for their objects. In educational platforms, this translates to content authors controlling who may download or comment on their materials. DAC is flexible but vulnerable to privilege escalation – a malicious user may grant excessive permissions to colluding accounts.

Mandatory Access Control (MAC) assigns security labels to both subjects and objects; access is permitted only when labels match according to a central policy. While MAC provides strong security guarantees (used in military systems), its rigidity makes it unsuitable for dynamic educational environments where content categories evolve continuously [1].

Role-Based Access Control (RBAC) defines permissions at the role level rather than the individual user level. Users are assigned to roles, and roles are assigned to permissions. RBAC is the dominant model in enterprise and educational systems due to its intuitive alignment with organizational hierarchies and ease of administration [5]. An instructor role, for instance, inherits all student permissions plus the ability to mark materials as official course resources.

Attribute-Based Access Control (ABAC) extends RBAC by evaluating contextual attributes – user department, time of access, resource sensitivity, or ownership – to make fine-grained access decisions. ABAC enables policies such as: ‘a student may rate a material only if they are not its author’. Such contextual rules cannot be expressed in pure RBAC [2]. Table 1 summarizes the comparative characteristics of these models.

Table 1 – Comparison of access control models for educational information systems

Model	Flexibility	Security	Scalability	EIS Fit
DAC	High	Low	High	Partial
MAC	Low	Very High	Low	Poor
RBAC	Medium	High	Medium	Good
ABAC	Very High	High	Very High	Excellent

Role model for educational platform

The design of access control for a peer-to-peer educational platform requires careful identification of user roles and their associated permission sets. Based on the principles of stakeholder analysis and the least-privilege model, three principal user roles can be defined for a typical student-oriented learning materials exchange system: Student, Instructor, and Administrator [6]. Each role encapsulates a distinct and non-overlapping set of capabilities, from content browsing and uploading to official content endorsement and system-wide moderation.

The Student role represents the core user class and combines both content consumption and content production capabilities. Students may search and filter the material catalog, preview and download files, upload their own materials with required metadata (title, description, subject, course, tags), edit or delete their own publications, rate materials on a 1–5 star scale, and subscribe to topic notifications. A critical business rule enforced at this level is the authorship check: a student cannot rate or report their own materials, preventing reputation manipulation. The student role implements the principle of least privilege – no write access to other users’ content and no access to administrative functions.

The Instructor role inherits all Student permissions and adds the privilege of marking materials with an ‘Official Course Material’ badge – a designation that increases trust visibility in search results. Instructors additionally have access to download analytics for their own materials, enabling insight into student engagement patterns across their courses. Instructor status is verified through university email domain matching during OAuth authentication.

The Administrator role operates outside the standard permission hierarchy. Administrators can delete any material, block any user account, access the full moderation journal, and view platform-wide analytics. All administrator actions are logged in an immutable audit trail – an ABAC attribute that prevents retroactive falsification of moderation decisions. The complaint escalation rule is also enforced at this level: when more than three complaints are filed against a single author, the system automatically generates a priority notification for the administrator. Table 2 presents the complete permission matrix for all three roles.

Table 2 – Role permission matrix for the educational platform

Permission	Student	Instructor	Administrator
Search & download materials	✓	✓	✓
Rate materials (not own)	✓	✓	✓
Upload & publish materials	✓	✓	–
Edit/delete own content	✓	✓	–
Mark as Official Course Mat.	–	✓	–
View own material analytics	✓	✓	✓
Subscribe to topic alerts	✓	✓	–
Moderate all platform content	–	–	✓
Block users & manage accounts	–	–	✓
View full audit log	–	–	✓

Hybrid RBAC+ABAC implementation

Pure RBAC cannot express contextual constraints such as the authorship-based rating prohibition or the complaint-based content freeze. These policies require attribute evaluation at request time. The recommended approach is a hybrid model: RBAC defines the structural permission baseline, while ABAC policies evaluate contextual attributes to refine access decisions [2].

For the educational platform, ABAC policies are implemented as server-side middleware interceptors. Before executing any write operation, the system evaluates a policy set in a JSON-based Policy Decision Point (PDP). For example, the rating policy simultaneously checks three attributes: `user.id ≠ material.authorId` (authorship constraint), `material.status = 'published'` (lifecycle constraint), and `user.role ∈ {'student', 'instructor'}` (role constraint). Only if all conditions are satisfied does the operation proceed [5].

The complaint-freeze mechanism demonstrates a dynamic ABAC attribute: when a user files a complaint against a material, the system sets a "frozen" status flag on that resource. This attribute is evaluated by the deletion policy – regardless of the requesting user's role, deletion is blocked for any frozen resource. This design resolves a common contradiction in peer-generated content platforms between the right to delete one's own content and the need to preserve disputed materials during active complaint processing [6]. The conflict between these two requirements is typically not apparent from text-based specifications alone and is only discovered through dynamic UML sequence diagram modeling.

OAuth 2.0 integration adds a delegation layer to the access model. When users authenticate via Google or Microsoft, the platform maps the OAuth-provided email domain to a university affiliation attribute, subsequently used in ABAC policies to restrict instructor badge privileges to verified university accounts, preventing unauthorized privilege escalation by non-academic users.

Security requirements and threat mitigation

The non-functional security requirements directly derive from threat modeling. The primary threats identified for the platform are: privilege escalation (a student invoking instructor-only API endpoints), brute-force credential attacks, cross-site scripting through material metadata fields, and malicious file uploads disguised as educational content [7].

Countermeasures are implemented at multiple layers. Transport security (TLS 1.2+) prevents credential interception. Bcrypt password hashing with cost factor ≥ 12 makes offline brute-force computationally infeasible. Rate limiting with account lockout after 5 failed attempts neutralizes online brute-force attacks. Input sanitization and Content Security Policy (CSP) headers prevent XSS exploitation of metadata fields. Antivirus scanning of all uploaded files prior to publication blocks malware distribution through the platform's file hosting capability. Together, these measures address the OWASP Top 10 vulnerabilities most relevant to educational content platforms [7]. It is critical that security requirements are expressed with measurable

verification criteria (e.g., SSL Labs score $\geq$ A, OWASP ZAP scan returning 0 critical findings) to enable objective compliance auditing rather than subjective assessment [8].

Conclusions

This research paper has examined the organization of user access rights management in educational information systems through theoretical analysis and applied requirements engineering. The following key findings were established.

A three-tier role hierarchy (Student, Instructor, Administrator) aligns naturally with academic organizational structures and supports the principle of least privilege at each level. The Student role consolidates content consumption and production capabilities into a single class, simplifying administration while preserving all necessary behavioral constraints through ABAC rules. A hybrid RBAC+ABAC model is the most suitable architecture for educational platforms with peer-generated content: RBAC establishes role-based permission baselines, while ABAC enforces contextual constraints such as authorship checks and dynamic content freeze states. Requirements modeling through UML diagrams is essential for discovering implicit access control rules that text-based requirements miss – the complaint-freeze contradiction was only identified during UML sequence diagram analysis. Non-functional security requirements must be specified with measurable metrics to enable objective verification during testing and audit.

Future research directions include federated identity management for cross-institutional access (SAML 2.0 / OpenID Connect), dynamic consent management for GDPR compliance, and the application of anomaly detection in user access patterns within educational platforms.

REFERENCES

1. Sandhu R. S., Coyne E. J., Feinstein H. L., Youman C. E. Role-Based Access Control Models. IEEE Computer. 1996. Vol. 29, No. 2. P. 38–47.
2. Hu V. C., Ferraiolo D. F., Kuhn D. R., Schnitzer A., Sandlin K., Miller R., Scarfone K. Guide to Attribute Based Access Control (ABAC) Definition and Considerations. NIST Special Publication 800-162. Gaithersburg: NIST, 2014. 54 p.
3. EDUCAUSE. 2023 EDUCAUSE Horizon Report: Teaching and Learning Edition. Boulder, CO: EDUCAUSE, 2023. 58 p.
4. Lampson B. W. Protection. In: Proceedings of the 5th Princeton Conference on Information Sciences and Systems. Princeton University, 1971. P. 18–24.
5. Ferraiolo D. F., Sandhu R., Gavrila S., Kuhn D. R., Chandramouli R. Proposed NIST Standard for Role-Based Access Control. ACM Transactions on Information and System Security. 2001. Vol. 4, No. 3. P. 224–274.
6. Alotaibi S. J., Roussinov D. Adaptive User Authentication Using Multi-Factor and Role-Based Mechanisms for Educational Information Systems. International Journal of Advanced Computer Science and Applications. 2022. Vol. 13, No. 4. P. 112–121.
7. OWASP Foundation. OWASP Top Ten 2021: The Ten Most Critical Web Application Security Risks. URL: <https://owasp.org/www-project-top-ten/> [Accessed: March 11, 2026].
8. Wiegers K., Beatty J. Software Requirements. 3rd ed. Redmond, WA: Microsoft Press, 2013. 672 p.

Лавренюк Арсен Олександрович – студент групи ІПІ-22б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: arsenlavreniuk@gmail.com.

Майданиук Володимир Павлович – кандидат технічних наук, доцент, доцент кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, e-mail: maidaniuk2000@gmail.com.

Магас Людмила Миколаївна – ст. викладач кафедри іноземних мов, Вінницький національний технічний університет, м. Вінниця, e-mail: magas@vntu.edu.ua.

Arsen O. Lavreniuk – student of group 1PI-22b, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: arsenlavreniuk@gmail.com.

Volodymyr P. Maidaniuk – Ph.D., Associate Professor of Software Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: maidaniuk2000@gmail.com.

Liudmyla M. Magas – FLD senior lecturer in English, Vinnytsia National Technical University, Vinnytsia, e-mail: magas@vntu.edu.ua.