UNIVERSITATEA "AUREL VLAICU" DIN ARAD
FACULTATEA DE ŞTIINŢE EXACTE
Department of Mathematics and Computer Science | Academic Year 2026–2027

Proposed Topics for Bachelor's Thesis (Licenţă)

Specializations: Computer Science / Mathematics

Advisor: Vlad-Florin DRAGOI

Overview & Guidance

The following thesis proposals are designed for undergraduate students seeking to specialize in theoretical and applied cryptography, system-level Linux/Unix security, web security, or error-correcting codes. Students are expected to combine mathematical foundations with practical software implementations.

Topic 1: Implementation and Performance Analysis of Code-Based Public Key Cryptosystems

Category: Post-Quantum Cryptography & Coding Theory
Description: McEliece public-key cryptosystems remain among the most resilient candidates against quantum computing attacks. This topic focuses on implementing classic McEliece variants using Goppa codes, analyzing key generator sizes, encryption/decryption throughput, and error-correction bounds.

- **Key Tasks:** Implement binary Goppa code generation and syndrome decoding; measure matrix generation complexity; compare security margins.
- **Prerequisites:** Abstract Algebra, C/C++ or Python, Basics of Information Theory.

Topic 2: Construction and Polarization Properties of Polar Codes

Category: Coding Theory & Information Theory
Description: Polar codes, introduced by Arikan, are the first class of constructive codes that provably achieve symmetric channel capacity. The student will investigate the channel polarization effect using Kronecker powers of generator matrices and implement Successive Cancellation (SC) decoding.

- **Key Tasks:** Simulate polarization over BEC/BSC channels; implement SC and SC-List (SCL) decoders; evaluate Frame Error Rate (FER).
- **Prerequisites:** Probability Theory, Linear Algebra, MATLAB/Python.

Topic 3: Hardening Unix/Linux Systems: PAM Modules and Kernel Security Modules

Category: Unix-Based System Security
Description: Unix system integrity relies on modular access controls and core authentication architecture. This project investigates system defense mechanisms through custom Pluggable Authentication Modules (PAM) and Linux Security Modules (LSM / AppArmor).

- **Key Tasks:** Develop a custom PAM module enforcing multi-factor or contextual access; configure strict AppArmor profiles; analyze dynamic binary security mechanisms.
- **Prerequisites:** Linux Systems Programming, C/Bash, POSIX Security Standards.

Topic 4: Auditing and Securing Modern Web Applications Against Injection and CSRF

Category: Web Application Security
Description: Web application security requires proactive defensive coding and rigorous vulnerability auditing. This topic covers the analysis and mitigation of OWASP Top 10 vulnerabilities, focusing on SQL Injection, Cross-Site Scripting (XSS), and Cross-Site Request Forgery (CSRF).

- **Key Tasks:** Build a vulnerable web laboratory; write automated security tests; implement robust Content Security Policy (CSP) headers and parameterized queries.
- **Prerequisites:** Web Technologies (HTML/JS/PHP or Python-Django/Flask), HTTP Protocol.

Topic 5: Conversions for Semantically Secure McEliece Encryption

Category: Public-Key Cryptography
Description: Original McEliece encryption is not semantically secure against chosen-ciphertext attacks (IND-CCA). This project examines padding schemes and conversion techniques (e.g., Kobara-Imai conversions) designed to achieve semantic security.

- **Key Tasks:** Study the Fujisaki-Okamoto and Kobara-Imai conversions; implement a secure wrapper over McEliece; evaluate performance overheads.
- **Prerequisites:** Cryptography Fundamentals, Modern Algebra, Python/SageMath.

Recommended Academic References

References

[1] R. J. McEliece, "A Public-Key Cryptosystem Based On Algebraic Coding Theory," *DSN Progress Report*, vol. 42, no. 44, pp. 114–116, 1978.

[2] N. Sendrier and R. Overbeck, "Code-Based Cryptography," in *Post-Quantum Cryptography*, D. Bernstein et al., Eds. Berlin, Germany: Springer, pp. 95–145, 2009.

[3] INRIA, "Code-Based Cryptography MOOC Course," Available online: https://www.canal-u.tv/producteurs/inria/cours_en_ligne/code_based_cryptography.

[4] A. Couvreur, "Algebraic Geometry Codes and Code-Based Cryptography," Slides PostScriptum, https://postscryptum.lip6.fr/couvreur.pdf.

[5] N. Sendrier, "Code-Based Cryptography and Post-Quantum Standardization," Video Lecture, https://www.youtube.com/watch?v=JThp--il3PI.

[6] K. Kobara and H. Imai, "Semantically Secure McEliece Public-Key Cryptosystems - Conversions for McEliece PKC," in *Proc. PKC '01*, LNCS 1992, Springer, pp. 19–35, 2001.

[7] E. Fujisaki and T. Okamoto, "Secure Integration of Asymmetric and Symmetric Encryption Schemes," in *Proc. CRYPTO '99*, LNCS 1666, Springer, pp. 537–554, 1999.

[8] D. Pointcheval, "Chosen-Ciphertext Security for Any One-Way Cryptosystem," in *Proc. PKC '00*, LNCS 1751, Springer, pp. 129–146, 2000.

[9] E. Arikan, "Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, 2009.

[10] I. Tal and A. Vardy, "How to Construct Polar Codes," *IEEE Trans. Inf. Theory*, vol. 59, no. 7, pp. 6562–6582, 2013.

[11] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*, Cambridge University Press, 2005.

[12] V. Samar and R. Scheimers, "Unified Login with Pluggable Authentication Modules (PAM)," in *Proc. 3rd ACM Conference on Computer and Communications Security (CCS)*, 1996.

[13] C. Wright, C. Cowan, S. Smalley, J. Morris, and G. Kroah-Hartman, "Linux Security Modules: General Security Support for the Linux Kernel," in *Proc. 11th USENIX Security Symposium*, pp. 17–31, 2002.

[14] S. Garfinkel, G. Spafford, and A. Schwartz, *Practical Unix and Internet Security*, 3rd ed., O'Reilly Media, 2003.

[15] OWASP Foundation, "OWASP Top 10 Web Application Security Risks," `https://owasp.org/www-project-top-ten/`, 2021.

[16] W. G. Halfond, J. Viegas, and A. Orso, "A Classification of SQL-Injection Attacks and Countermeasures," in *Proc. IEEE International Symposium on Secure Software Engineering*, pp. 13–15, 2006.

[17] D. Stuttard and M. Pinto, *The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws*, 2nd ed., John Wiley & Sons, 2011.