CyberMUG: Cybersecurity Modules aligned with UG Computer Science and Engineering Curriculum

PIs: Ahmad Y Javaid[1], Quamar Niyaz[2], Xiaoli Yang[2], and Sidike Paheding[3]

Student Researchers: Sai S Sudha[1], Jyothirmai Bandreddi[1], Bhanu Cherukuri[1], Kenny Guernsey[2], Jansen Tan[2], Jacob Tietz[2], Mateo Garcia[2], Gabe Silva[2], Divya Ravindra[2], and Abel Reyes[3]

[1]The University of Toledo, Toledo, Ohio, [2]Purdue University Northwest, Hammond, Indiana, [3]Michigan Technological University, Houghton, Michigan

Award # 2021264 and 2021345

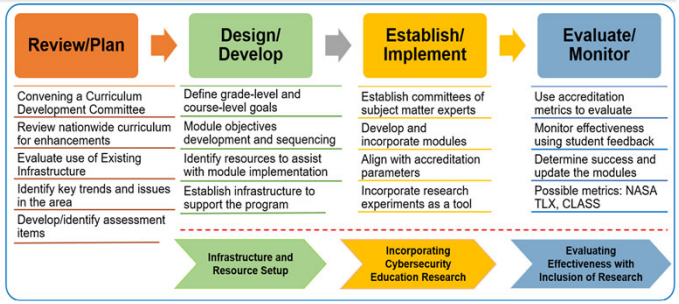
GOALS

- Introduce cyber-secure design and development practices to minimize vulnerabilities during system design rather than the addition of security measures as an after-thought
- Develop a security mindset at the undergraduate level with a focus on learning how intertwined cybersecurity is with any computation related activity/work/job
- Inculcate interest in cybersecurity careers with a long-term goal of addressing the national shortage of cybersecurity skills.

RESEARCH QUESTIONS

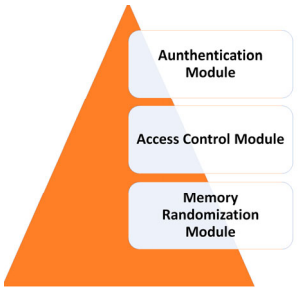
1. What courses in the curriculum are most effective for cybersecurity?
   a) What aspects of such modules capture students' interest?
   b) Do these modules help students grasp the role of cybersecurity in particular CE/CSE courses?
2. Does the addition of these modules improve student participation?
   a) What are the effects of these modules on cybersecurity learning
   b) Does the integration of cybersecurity modules increase students' interest in pursuing cybersecurity as a career?

CYBERMUG PLAN

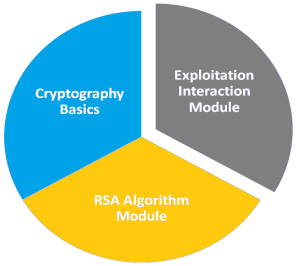


METHODOLOGY

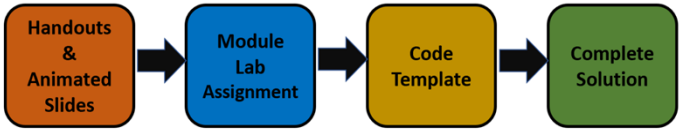
Course	Tool/Module Developed
Discrete Structures	Cryptography and RSA Algorithm Visualization tool
Operating Systems	Mini Project-based module using xv6
Software Engineering	Implementing security in SDLC
Intro to Programming	Secure Programming in C++ & Java through Practice Oriented Modules



Modules of Operating Systems and Systems Programming



RSA Algorithm Visualization Modules Discrete Structures

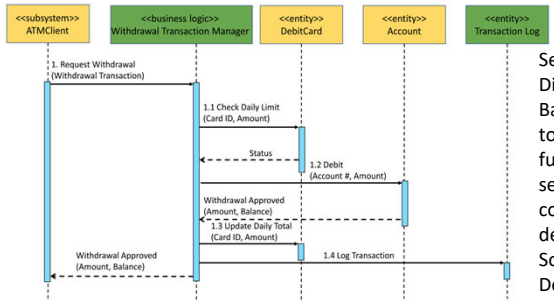


Various components of cybersecurity modules developed for C++/Java in Intro to Programming course.

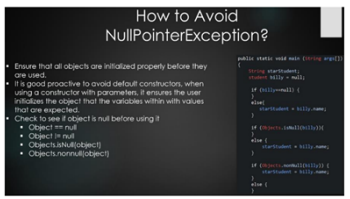
SDLC Phases	Regular tasks	Secure software tasks
Requirements	• Functional Requirements • Use cases	• *Security Requirements* • *Abuse cases*
Design	• Static and Dynamic Modeling	• *Architectural Risk Analysis* • *Security-oriented Design*
Implementation	• Coding	• *Code review*
Testing	• Functional Testing	• *Risk-based Security Tests* • *Penetration Testing*

SDLC phases integrated with security-related tasks in Software Design

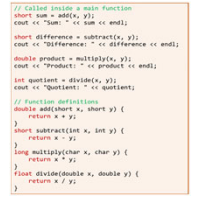
RESULTS



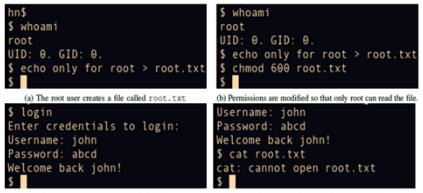
Sequence Diagram of Banking service to withdraw funds with secure component developed in Software Design



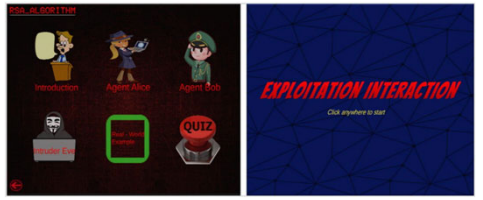
Null pointer module for Java in Intro to Programming course



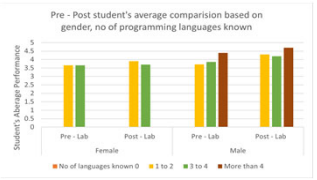
Code snippet for parameter mismatch module for C++



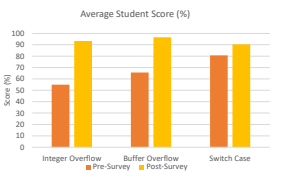
Implementation of the access control module in xv6 of Operating Systems and Systems Programming course



Example menu screens in the RSA/Cryptography tool



Pre/post improvement using the RSA Cryptography module



Pre/post improvement using secure programming module

CONCLUSION / FUTURE WORK

- More case studies will be added to the framework of Software Engineering and usability be assessed by students
- Java/C++ modules to be revised in response to student feedback
- Behavior of implementation of modules in xv6 found to be relatively similar to OS such as Linux, although differ in the depth of execution.
- Discrete Structures module is well received by students, as indicated by post-survey results.
- Further work on assessment and promotion to other institutes for use in their CS/CSE/CE courses.