CodeHS

Indiana Computer Science III: Cybersecurity Syllabus
High School (120-155 contact hours)

Course Overview and Goals

In this course, students are introduced to the secure software development process including designing secure applications, writing secure code designed to withstand various types of attacks, and security testing and auditing. It focuses on the security issues a developer faces, common security vulnerabilities and flaws, and security threats. The course explains security principles, strategies, coding techniques, and tools that can help make software fault-tolerant and resistant to attacks. Students will write and analyze code that demonstrates specific security development techniques. Students will also learn about cryptography as an indispensable resource for implementing security in real-world applications. Students will learn the foundations of cryptography using simple mathematical probability. Information theory, computational complexity, number theory, and algebraic approaches will be covered.

Learning Environment: The course utilizes a blended classroom approach. The content is a mix of web-based and physical activities. Students will modify existing code and run it in the browser, investigate cyber related topics and reflect on them and discuss them, create digital presentations, and engage in in-person collaborative exercises with classmates. Teachers utilize tools and resources provided by CodeHS to leverage time in the classroom and give focused 1-on-1 attention to students.

Programming Environment: Students modify and run programs in the browser using the CodeHS online editor. Students will be able to modify text-based programs in HTML, SQL and simulate shell commands. Students will also participate in simulated cyber attacks on safe sites in order to learn how to mitigate cyber attacks. Students will be able to document their processes and discuss best practices for preventing cyber attacks.

Quizzes: Each lesson includes at least one formative short multiple choice quiz. At the end of each module, students take a summative multiple choice quiz that assesses their knowledge of the concepts covered in the module.

Recommended Grade Level: 11, 12
Required Prerequisite: Indiana Computer Science I and Indiana Computer Science II

More information: Browse the content of this course at https://codehs.com/course/10265

Course Breakdown

Module 1: What is Cybersecurity? (1-2 weeks/5-10 hours)
This module provides an introduction to cybersecurity. It focuses on why cybersecurity is important, recent threats to cybersecurity, and different careers in the field.

Browse the full content of this module at https://codehs.com/library/course/10265/module/15439

Objectives / Topics Covered	• Course Overview • What is Cybersecurity? • Impact of Cybersecurity • The CIA Triad

Example Assignments / Labs	• Course Overview
	  ○ Do you use the Internet?
	  ○ How do you use the Internet?
	  ○ What kinds of information are at risk?
	  ○ What are some different CS career fields?
	  ○ Coding as the new literacy
	  ○ What is this course about?
	  ○ Example activity:
	  ■ Lists steps to take to protect yourself on the Internet
	  ■ What is something you want to know or make by the end of the course?
	• What is Cybersecurity?
	  ○ Cybersecurity defined
	  ○ Why is cybersecurity important?
	  ○ Cybersecurity in the news
	  ○ Cybersecurity and IoT (Internet of Things)
	  ○ How do we prevent cyber attacks?
	  ○ Example activities:
	  ■ Summarize and discuss recent cyber attacks
	  ■ Explore a threat map to see where cyber attacks are coming from and which countries are being targeted
	• Impact of Cybersecurity
	  ○ Why do we care about cybersecurity?
	  ○ What information is at risk?
	  ○ What are the impacts of cyber attacks?
	  ■ Financial impact
	  ○ Cybersecurity workforce
	  ○ What are current cybersecurity career?
	  ○ Example activities:
	  ■ Review resources and reflect on or discuss
	  • What information do cyber criminals steal?
	  • What do cyber criminals do with stolen information?
	• The CIA Triad
	  ○ What is the CIA triad? (confidentiality, integrity, availability)
	  ○ What are “secure systems?”
	  ○ What do confidentiality, integrity, and availability mean in cybersecurity?
	  ○ Example activities:
	  ■ Determine where scenarios break part of the CIA Triad

Module 2: The ABCs of Cryptography (1-2 weeks/5-10 hours)
In this module, students will dive into the history of cryptography systems, the motivation behind using encryption systems, and basic cryptography systems. Additionally, they will explore topics on how to use cryptography, cryptology, and cryptanalysis to decode a message without the use of a key.

Browse the full content of this module at https://codehs.com/library/course/10265/module/15443

Objectives / Topics Covered	• Cryptography, Cryptology, Cryptanalysis • History of Cryptography • Why do we Need to Encrypt Data? • Basic Cryptography Systems: Caesar Cipher • Basic Cryptography Systems: Cracking the Caesar Cipher • Basic Cryptography Systems: Vigenère Cipher

Example Assignments / Labs	• Cryptography, Cryptology, Cryptanalysis
	  ○ Why do we need some secrecy in our transparent information age?
	  ○ Explain general encryption with data, keys
	  ○ Example activities:
	  ■ Video and discussion on securing the cloud
	  ■ Passing notes in class (offline activity)
	• History of Cryptography
	  ○ Why do we encrypt?
	  ○ What are some classic encryption techniques?
	  ○ What is the flaw in substitution ciphers?
	  ○ What was The Enigma during WW2?
	  ○ What is modern cryptography and how has cryptography changed over time?
	  ○ What is 256-bit key encryption and how does this help cryptography overall?
	  ○ Example activities:
	  ■ How did the Enigma work?
	• Why do we Need to Encrypt Data?
	  ○ Explore the CIA Triad and encryption
	  ○ Example activities:
	  ■ Telephone game with math (offline)
	  ■ Modulo math activity sheet
	• Basic Cryptography Systems: Caesar Cipher
	  ○ Explore examples of the Caesar cipher
	  ○ Example activities:
	  ■ Practice with a Caesar Cipher JavaScript program
	  ■ Modify the program to create the decrypting Caesar program
	• Basic Cryptography Systems: Cracking the Caesar Cipher
	  ○ How do we solve the Caesar Cipher with brute force and using letter frequency analysis?
	  ○ Example activities:
	  ■ Practice cracking Caesar Cipher with brute force
	  ■ Practice cracking Caesar Cipher with letter frequency
	• Basic Cryptography Systems: Vigenère Cipher
	  ○ Explore examples of the Vigenère Cipher
	  ○ Example activities:
	  ■ Practice with a Vigenère Cipher JavaScript program

Module 3: Project: Classic Cipher Newscast (1 week/5 hours)
Students complete a project to apply cryptography content.

Browse the full content of this module at https://codehs.com/library/course/10265/module/15444

Objectives / Topics Covered	● Project: Classic Cipher Newscast
Example Assignments / Labs	● Project: Create a Newscast ○ Students work collaboratively to research a **classic cipher** (beyond Caesar and Vigenere) to address in their newscast. They will investigate their cipher and write a script that includes how the cipher works, when it was used, and when the cipher stopped being useful.

Module 4: Advanced Cryptography (4 weeks/20 hours)
Students will apply advanced principles of cryptology. This includes explaining the core concepts of Public Key Infrastructure and hash functions. Students will explore concepts of encrypted email, digital certificates, and private key certificates. They will understand the different types of SSL certificates, the chain of trust and how a Certificate Authority (CA) works.

Browse the full content of this module at https://codehs.com/library/course/10265/module/15468

Objectives / Topics Covered	● Encryption Algorithms ● Public Key Encryption ● Hash Functions ● Asymmetric Encryption ● Digital Certificates
Example Assignments / Labs	● Encryption Algorithms ○ What are the key functions of cryptography? ○ What is a block cipher? ○ How many bits are used in each block in the Data Encryption Standard (DES)? ○ How does the Advanced Encryption Standard (AES) compare with the DES? ○ Example activity: ■ What is an advantage of using a key instead of a random substitution? ■ Using the Rail Fence Cipher, encrypt your own message and trade with a partner. See if you can decrypt the message without knowing how many rails your partner used. ■ Is the Pigpen cipher stronger than the Caesar and Mixed Alphabet cipher? Why or why not? ● Public Key Encryption ○ What are the differences between symmetric and asymmetric encryption? ○ What happens during public key encryption? ○ Example activity: ■ What is REALLY meant by “keys” in the computing world? ■ What kind of number procedure do you need to have to make it impossible for Eve to determine any message sent

	between Alice and Bob? ● Hash Functions ○ What is a collision in a hash function? ○ What is password salting? ○ How does modulo math increase the strength of an encryption? ○ Example activity: ■ Why must each “salt” be unique for each password? ■ Develop a simple hash function by changing the math in the function createHash(). Be sure to keep some kind of modulo in your math, so there’s no easy way to calculate information based on the types and quantities of certain characters in any message. ● Asymmetric Encryption ○ Man-in-the-middle attacks affect which part of the CIA triad? ○ What is a vulnerability of the Diffie-Hellman’s key exchange? ○ Example activity: ■ How is a trapdoor function used in the Diffie-Hellman key exchange? How is this related to RSA encryption? ■ What is OpenPGP? ● Digital Certificates ○ What are the different types of SSL certificates? ○ What is the maximum SSL Certificate duration of validity? ○ What is the chain of trust? ○ How can certificate pinning and stapling help prevent man-in-the-middle attacks? ○ Example activity: ■ Connection: How is using a notary public similar to the use of SSL certificates? ■ Become a Certificate Authority: Create a flyer, commercial, or advertisement promoting your certificate authority service.

Module 5: Project: Steganography (1 week/5 hours)
Students will explore steganography and create their own encryption algorithm to conceal and hide a message within the pixels of an image.

Browse the full content of this module at https://codehs.com/library/course/10265/module/15472

Objectives / Topics Covered	● Steganography ● Data Hiding and Extraction ● Encryption Algorithms
Example Assignments / Labs	● Hide a message! Students will create their own pixel picture using a web-based tool to hide a message in using the tool. They will change the hexadecimal values just slightly according to an encryption algorithm that they have created to hide their message!

Module 6: System Administration (3-4 weeks/15-20 hours)
Students will compare and contrast common operating systems (Windows, Linux, OS) and explain the importance of application security. They will investigate security options and implement user accounts to enforce authentication and authorization. Students will also demonstrate how to work with basic and advanced command prompts.

Browse the full content of this module at https://codehs.com/library/course/10265/module/15445

Objectives / Topics Covered	• Operating Systems • Software and Applications • Application Security • Browser Configuration • System Administration • Command Line Interface

Example Assignments / Labs	• Understanding Operating Systems
	• Comparing Operating Systems
	  ○ Installing an OS
	• File Management
	  ○ What Processor are you Running?
	• Software Licenses
	• Antivirus Software
	  ○ Data Backups
	• Using Cache
	• Popup Blockers
	• User Accounts
	  ○ Admin vs. Standard
	• Host Security
	  ○ Using a Log
	• System Commands
	  ○ cd, ls, mk etc
	• Network Commands
	  ○ ipconfig, netstat etc

Module 7: IT Infrastructure (2-3 weeks/10-15 hours)

Students will learn about the physical elements of computers and networking such as motherboards, RAM, routers, and the use of port numbers, ethernet and wireless devices.

Browse the full content of this module at https://codehs.com/library/course/10265/module/15449

Objectives / Topics Covered	• Internal Components of a Computer • Peripheral Devices • Network Devices • Storage and Network Options • Network Communication • Network Management

Example Assignments / Labs	• Different Types of CPU
	• RAM vs. Hard Drive
	• Wireless Internet Connections
	  ○ Speed Test
	• Security of Cloud Storage
	• Ethernet Standards
	• Setting Up a Firewall
	  ○ Establish Firewall Rules
	• SSH Logs
	  ○ Reading Logs

Module 8: Project: Troubleshooting Project (1 week/5 hours)
Students will explore the troubleshooting methodology and utilize it to solve sample IT support issues.

Browse the full content of this module at https://codehs.com/library/course/10265/module/15450

Objectives / Topics Covered	● Troubleshooting Methodology ○ Identify the problem ○ Research past solutions ○ Establish a theory ○ Test the theory ○ Establish a plan of action ○ Implement the solution ○ Verify functionality ○ Document findings
Example Assignments / Labs	● Troubleshooting: In this project, students will learn more about each step of the troubleshooting methodology and use these steps to repair and improve faulty network systems. ○ Poor Signal Strength ○ Interference

Module 9: Software Security (3-4 weeks/15-20 hours)
In this module, students will learn what happens when running a web application and how to look inside web apps using developer tools, source code, and more. They will learn basic SQL and common attacks like SQLi. Students will also be able to recommend solutions for flawed security systems.

Browse the full content of this module at https://codehs.com/library/course/10265/module/15446

Objectives / Topics Covered	● Inside Web Applications ● Developer Tools ● The Value of Data ● SQL Overview ○ What is SQL? ○ Structuring Data in SQL ○ Basic Querying in SQL ○ Filtering Queries in SQL ● Clients, Servers, Databases ● Common Security Problems ● SQL Injection ○ SQLi Overview ○ Types of SQLi ○ Preventing SQLi
Example Assignments / Labs	● Inside Web Applications ○ View page source (images, navigation and page layout, stylesheets, JavaScript, minified code ○ Example activities: ■ View page source scavenger hunt ■ Getting started with OWASP ● Developer Tools ○ Use the inspect tools to look more deeply inside of web apps

	○ How does view page source compare to inspect in terms of information about the site / app? ○ Example activities: ■ Practice using the Chrome developer tools ■ Change a favorite site using the Chrome developer tools on your end only. Take a screenshot of your change. ● Data Visualizations ● Design a Survey ● SQL Overview ○ What is SQL? ○ How do we structuring data using SQL? ○ How do we query databases using SQL? ○ Example activities: ■ Use the SELECT statement to query a database ■ Use the WHERE clause to query a database ● Clients, Servers, Databases ● Common Security Problems ○ What is the “Fortification Principle”? ○ What are some tips about HTTP vs. HTTPS, password fields and CAPTCHA that can help us to navigate more securely on the Web? ● SQL Injection ○ SQLi Overview ■ What is SQLi? ■ Why is SQLi a problem? ■ What happens during a SQLi attack? ■ What is the the fallout of a SQLi attack? ■ How does SQLi work? ■ How do hackers use SQL in a SQLi? ○ What are the types of SQLi (error-based, union-based, blind) ■ What is the underlying SQL behind the scenes that hackers may be trying to hack? ○ How to we mitigate or prevent SQLi? ■ What are the OWASP recommendations? ■ How can we tell if our code is vulnerable? ○ Example activities: ■ Discuss the Equifax SQL injection attack ■ Practice basic SQLi on a safe site ■ Research SQLi prevention

Module 10: Project: Security Assessment Report (1 week/5 hours)
Students complete a project that has them test a website for vulnerabilities and write a security assessment report based on their findings.

Browse the full content of this module at https://codehs.com/library/course/10265/module/15447

Objectives / Topics Covered	● Project: Security Assessment Report
Example Assignments / Labs	● Project: Security Assessment Report ○ SQLi Testing ○ Create a Security Assessment Report ○ Project Reflection

Module 11: Software Development Lifecycle (2-3 weeks/10-15 hours)
In this project, students will develop a training policy that informs employees on matters of network security and details the company policy on preventative measures employees should take.

Browse the full content of this module at https://codehs.com/library/course/10265/module/15470

Objectives / Topics Covered	● User Training ● Incident Response Plans ● Data Policy and Privacy ● Change Management
Example Assignments / Labs	● Develop a training policy that informs employees on matters of network security. ● Create an Incidence Response Plan. ● Develop a strong data policy for a company. ● Develop a change management plan to ensure that the new policy is adopted and implemented by the team effectively.

Module 12: Risk Management (4 weeks/20 hours)
Students will demonstrate skills in conducting vulnerability scans and recognizing vulnerabilities in security systems. They will conduct a security audit and examine port scanning, packet sniffing, and proxy servers to discover exploits in a system. Students will recommend security measures to mitigate the vulnerabilities.

Browse the full content of this module at https://codehs.com/library/course/10265/module/15469

Objectives / Topics Covered	● Identifying Risks ● Assessing Risks ● Risk Response ● Penetration Testing
Example Assignments / Labs	● Identifying Risks ○ What are the steps of a risk assessment? ○ What potential risks can be checked by a vulnerability scan? ○ How is packet sniffing and password cracking used in a legal manner? ○ Example Activity: ■ What information can be determined by an IP address? ■ Create a “story” using the data shown of what was happening during this packet transfer. ■ Why is past data important in trying to access how to best set up a cyber defense system for the present? ● Assessing Risks ○ What is a race condition? ○ What is error handling and input handling? Why is input validation important? ○ What is buffer overflow and integer overflow? ○ Example Activity: ■ Draft an argument that insists upon the importance of upgrading a system that has reached its end-of-life. ■ Read a scenario and access the level of risk. ■ Examine (and fix) poor input and error handling. ● Risk Response

	○ What are some risk response strategies? ○ How do you calculate the SLE and ALE of a threat event? ○ How do you effectively and efficiently mitigate risk? ○ Example activity: ■ Read a sample assessment report. What types of methods did the assessors use to collect data? Do you feel this report provides you with sufficient information to determine priorities and next steps? ■ What role might chaos engineering play in risk assessment and response? ● Penetration Testing ○ What are the stages of penetration testing? ○ What tools are used in passive reconnaissance? ○ What is an escalation of privilege? ○ Example activity: ■

Module 13: Project: The Game of Risk (2-3 weeks/10-15 hours)

In this project, students will design and create a board game or a card game that will help players to identify randomized security vulnerabilities and their appropriate defenses.

Browse the full content of this module at https://codehs.com/library/course/10265/module/15473

Objectives / Topics Covered	● Quantitative and Qualitative SLE ● Prototypes ● Testing
Example Assignments / Labs	● Create a Game: Students will design and create a board game that will help players to identify randomized security vulnerabilities and their appropriate defenses. They will create a prototype and test the game to receive feedback to consider before building their final game.