Syllabus:  
Secure development

**Course name:** Secure development

**Code discipline:** XXX

**Subject area:** Security and Networks

## P.1 Short Description

The course is aimed to cover security aspects of development. It covers security architecture, secure coding, security assurance, security operation and basic security concepts. We would go from the deepest kernel (ASLR, NX/DEP, CET, KPTI against ROP, UAF, etc) to theoretical high (access models, Biba and Bell-LaPadula model, security principles), from practical design (NIST recommendations and security by design) to day-by-day operations (OSA practices). We would discuss fuzzing, stat analyzers power and SIEM. The course would be extremely useful for security architects. We would discuss not only security, but also safety topics because the mitigations for them are intersecting. The examples would be given based on Linux OS.

**P.2 Prerequisites**

What subjects and topics students should have covered before starting the course to succeed.

### P.2.1 Prerequisite subject from [the list](https://docs.google.com/spreadsheets/d/1LGrcNu_mAeOr9K2QehDgHSVrRL5P2IcaGFOvQqEaGUE/edit#gid=0)

* CSE101: Introduction to Programming
* CSE112: Software Systems Analysis and Design
* CSE105 or CSE128 or CSE130

### P.2.2 Prerequisite topics

* Basic programming skills, C/C++ is recommended
* Software design or software architecture
* Basics of compilers
* Basics of computer architecture (Intel or ARM is preferrable)

## P.3 Course Topics

**Table 1:** Course Sections and Topics

|  |  |
| --- | --- |
| **Section** | **Topics within the section** |
| 1. Basics of security | * Security and safety. Security and code quality. Maintainability and security. Why it is so hard to develop a secure system and what approaches may be applied? When it makes sense to drive system secure? |
| 1. Security architecture | * NIST recommendations * Security principles * Theoretical security: access matrix and security models * Secure by design |
| 1. Secure coding | * Security on the code level * SDL * Main binary vulnerabilities and their mitigations |
| 1. Secure operating | * Security monitoring * DevSecOps * Dealing with 3rd parties |
| 1. Security assurance | * Pen testing * Fuzzing * Bug Bounty programs |
| 1. Linux security | * Keep it all together and see how Linux kernel deals with that. * SELinux * GrSec patches * Why Linux is not safety system |

## P.4 Intended Learning Outcomes (ILOs)

### P.4.1 What is the main purpose of this course?

The main purpose of this course is to give students a security vision from up to down, because the security principle of weakest link insist that the weakest part of the process/system would be the one to be attacked.

### P.4.2 ILOs defined at three levels

We specify the intended learning outcomes at three levels: conceptual knowledge, practical skills, and comprehensive skills.

**Level 1:** What **concepts** should a student **know/remember/explain?**

By the end of the course, the students should be able to ...

* Reason about the limitation of different security policies
* Remember main security principles
* List SDL stages
* Describe the difference between security and safety
* Explain basic binary vulnerabilities
* Specify the required security assurance
* Describe the key elements of SOC systems
* Explain why fuzzing is not the same as unit or integration testing

**Level 2:** What **basic practical skills** should a student be able to perform?

By the end of the course, the students should be able to ...

* Read CVEs and understand its impact instead of trusting other experts
* Perform Threat Modeling
* Review code to find insecure patterns
* Deal with open source code securely
* Explain the value of bug bounty programme and find the right moment to start it

**Level 3:** What **complex comprehensive skills** should a student be able to **apply in real-life scenarios?**

By the end of the course, the students should be able to ...

* Reason about security and safety of the system
* Suggest hardenings and architecture drifts to achieve required level of s&s
* Propose process improvement in a cost-effective manner that would drastically improve the security and safety level.

## P.5 Grading

**Table 2:** Course grading range

|  |  |  |
| --- | --- | --- |
| **Grade** | **Range** | **Description of performance (optional)** |
| A. Excellent | 80-100 |  |
| B. Good | 60-79 |  |
| C. Satisfactory | 40-59 |  |
| D. Fail | 0-39 |  |

**Table 3:** Course activities and grading breakdown

|  |  |
| --- | --- |
| **Activity Type** | **Percentage of the overall course grade** |
| Assignment/Labs | 70 |
| Final quiz | 30 |

### P.5.1 Recommendations for students on how to succeed in the course (optional)

* Participation is important. Showing up is the key to success in this course.
* If you don’t have a corresponding technical background, please do not hesitate to ask lecturer. If you feel that the gap is deep, request for extra reading.
* Reading the recommended literature is optional, and will give you a deeper understanding of the material.

## P.6 Resources, literature and reference materials

### P.6.1 Open access resources

* Owasp.com
* MITRE SOC Operations https://www.mitre.org/sites/default/files/publications/11-strategies-of-a-world-class-cybersecurity-operations-center.pdf
* MISRA, AUTOSAR, SEI CERT
* <https://github.com/veeral-patel/how-to-secure-anything>
* <https://awesome-safety-critical.readthedocs.io/en/latest/>
* <https://www.iiconsortium.org/smm.htm>
* <https://www.cisa.gov/uscert/bsi/articles/knowledge/coding-practices>
* https://www.microsoft.com/en-us/securityengineering/sdl
* Managing Security Risks Inherent in the Use of Third-Party Components

<https://safecode.org/wp-content/uploads/2017/05/SAFECode_TPC_Whitepaper.pdf>

### P.6.2 Closed access resources

* Matt Bishop, (2018) “Computer Security: Art and Science”
* D Deougun, DB Jonhsson, D Sawano (2019) “Secure by design”
* D LeBlanc, Michael Howard (2002) “Writing secure code”
* ISO26262

### P.6.3 Software and tools used within the course

* Some static analyser
* AFL
* <https://docs.microsoft.com/en-us/azure/security/develop/threat-modeling-tool>
* snyk.io

Teaching Methodology:  
Methods, techniques, & activities

## P7. Activities and Teaching Methods Matrices

Mark what techniques, methods, and activities are used in each section (1 is used, 0 is not used).

**Table A1:** Teaching and Learning Methods within each section

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Teaching Techniques** | **Section 1** | **Section 2** | **Section 3** | **Section 4** | **Section 5** | **Section 6** |
| Problem-based learning (students learn by solving open-ended problems without a strictly-defined solution) | 1 | 1 | 1 | 1 | 1 | 1 |
| Modular learning (facilitated self-study) | 0 | 1 | 0 | 0 | 1 | 0 |
| Differentiated learning (provide tasks and activities at several levels of difficulty to fit students needs and level) | 1 | 1 | 1 | 1 | 1 | 1 |
| Contextual learning (activities and tasks are connected to the real world to make it easier for students to relate to them); | 0 | 1 | 1 | 0 | 0 | 0 |
| Business game (learn by playing a game that incorporates the principles of the material covered within the course). | 0 | 0 | 0 | 0 | 0 | 1 |
| Studio-based learning | 0 | 0 | 0 | 0 | 0 | 0 |

**Table A2:** Activities within each section

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Learning Activities** | **Section 1** | **Section 2** | **Section 3** | **Section 4** | **Section 5** | **Section 6** |
| Lectures | 1 | 1 | 1 | 1 | 1 | 1 |
| Lab exercises | 1 | 1 | 1 | 1 | 1 | 1 |

## P.8 Formative Assessment and Course Activities

Assessment and activities allow the instructor to check the students’ understanding and mastery of the course material, including theoretical and practical knowledge.

In this section you have two types of evaluation:

* Ongoing
* Practical Tasks for Labs

Assessment is about giving the students a grade based on their mastery of the subject. The two types of assessment you need to provide are:

* final assessment
* retake

### P.8.1 Ongoing performance assessment

Both graded and non-graded activities during the semester (before the exam)

#### Section 1

|  |  |  |
| --- | --- | --- |
| **Activity Type** | **Content** | **Is Graded?** |
| Individual Assignments | A2: Product Ideation and Market Research Find all weakness in the code snippet. Suggest how to fix them in a secure way. What is your recommendation for the code author? | 1 |

### P.8.2 Final exam questions

The quiz is open book but time limited. Here are some questions that could be on exam:

1. As a CISO you are going to formulate new security policy on printer use in the company. It is supposed that a printer is allowed for administrating only by IT helpdisk and printing is allowed only for secretaries and project managers. What kind of policy is it?

- DAC

- MAC

- RBAC