Syllabus

1. Program information

1.1. Institution	ACADEMY OF ECONOMIC STUDIES
1.2. Faculty	Economic Cybernetics, Statistics and Informatics
1.3. Departments	(Departament) INFORMATICA SI CIBERNETICA ECONOMICA
1.4. Field of study	Economic Informatics
1.5. Cycle studies	Master Studies
1.6. Education type	Full-time
1.7. Study program	IT&C Security
1.8. Language study	
1.9. Academic year	2017-2018

2. Course information

2.1. Name	**Source Code Programming Security**								
2.2. Code	**17.0241IF2.1-0001**								
2.3. Year of studies	**2**	2.4. Semester	**1**	2.5. Assessment type	**Exam**	2.6. Course type	**O**	2.7. Number of ECTS	**6**
2.8. Instructors									

3. Total estimated time

3.1. Number of weeks	14.00		
3.2. Number of hours per week	2.00	of which	
		C(C)	1.00
		S(S)	1.00
3.3. Total hours from curriculum	28.00	of which	
		C(C)	14.00
		S(S)	14.00
3.4. Total hours of study per semester (ECTS*25)	150.00		
3.5. Total hours of individual study	122.00		
Time distribution for individual study			
Study the textbook, course support, bibliography and notes			
Further reading in the library, on the online platforms and field			
Preparing seminars, labs, homework, portfolios and essays			
Tutoring			
Examinations			
Other activities			

4. Prerequisites

4.1. About curriculum	Secure Application Programming
4.2. About skills	C/C++ Programming

5. Requirements

C(C)	Course lectures take place in rooms with multimedia teaching equipment.
S(S)	Laboratories are held in rooms that have PCs with Internet access. The development environment used is Microsoft Visual Studio 2010 or 2012, Ubuntu within virtual machines with GCC, Java plus necessary tools.

6. Skills covered

	C3	Using modern computer technologies for developing components that ensure maximum IT security
	CT2	Planning and organization of human resources within a team or organization, in terms of awareness of the responsibility for professional results

7. Course objectives

7.1. General objective	Programming for secured informatic systems. Presentation of mechanisms, technologies and techniques for securing the source code.
7.2. Specific objectives	Transfer tehnologic pentru: -Stack overflow -Buffer overrun -Code injection

8. Course contents

8.1. C(C)		Teaching methods	Advices
1	1. Secure programming techniques : buffer overrun, stack overflow, obfuscating; plus Source code security – Overview		
2	2. Common mistakes: signed-unsigned, integer over / under - flow; Smashing the Stack. Heap Overrun taxonomy. Array Indexing Errors.		
3	3. Minimal privileges execution; Access Control.		
4	4. Ecryption integrity;		
5	5. Protection against DoS attacks - Denial of service.		
6	6. Code security testing;		
7	7. Programs documentation;		
8	8. File naming / working vulnerabilities. String function library vulnerability		
9	9. Race condition. Multithread programming security		
10	10. DB access security - Code Injection		
11	11. Vulnerabilities in Object Oriented Programming		
12	12. Protecting the code from reverse engineering. Obfuscators		
13	13. Virtual machine / environment vulnerabilities : .NET		

Bibliography

- Ross Anderson, Security Engineering, John Wiley & Sons, 2001
- Eric Rescorla, SSL and TLS: Designing and Building Secure Systems, Addison-Wesley, 2001
- Ion Ivan, Cristian Toma , Informatics Security Handbook - 2nd Edition, ASE Publishing House, 2009
- Stalling Willliam, Cryptography and Network Security, Prentice Hall, 1999
- Aleph One, Smashing The Stack For Fun And Profit, Phrack Magazine, vol. 49, Phrack Magazine, 1996
- Michael Howard, David LeBlanc, Writing Secure Code, Hard-copy Book, Microsoft Press, 2003, Statele Unite ale Americii

8.2. S(S)		Teaching methods	Advices
1	1. Secure programming techniques : buffer overrun, stack overflow, obfuscating;		
2	2. Access control;		
3	3. Minimal privileges execution;		
4	4. Ecryption integrity;		
5	5. Protection against DoS attacks;		
6	6. Code security testing;		
7	7. Programs documentation;		
8	8. File naming / working vulnerabilities. String function library vulnerability		
9	9. Race condition. Multithread programming security		
10	10. DB access security - Code Injection		
11	11. Vulnerabilities in Object Oriented Programming		
12	12. Protecting the code from reverse engineering. Obfuscators		
13	13. Virtual machine / environment vulnerabilities : .NET		

Bibliography

- Ion Ivan, Cristian Toma , Informatics Security Handbook - 2nd Edition, ASE Publishing House, 2009
- Eric Rescorla, SSL and TLS: Designing and Building Secure Systems, Addison-Wesley, 2001
- Ross Anderson, Security Engineering, John Wiley & Sons, 2001

9. Course contents corroboration with the demands of epistemic community representatives, professional associations and representative employers

Taking into account the best practices from IT&C field applied by big companies such as: Intel, Oracle, Microsoft, IBM, HP and professional consortiums such as: Apache, Red Hat, ISO/IEC.

10. Assessment

Activity	Assessment criteria	Assessment methods	Percentage in the final grade
10.1. S(S)		Applied activities, practical or project certificates/laboratory/tests, tests during the module, auditing tests	40.00
10.2. Final assessment		Final examination	60.00
10.3. Grading scale	Whole notes 1-10		
10.4. Minimum performance standard	Knowledge required: reservation / allocation and stack and heap memory management in oreder to avoid stack overflow or buffer overrun. Preventing common vulnerabilities in source code; mitigating the impact. The point granted by default is included in the weights assigned to the types of assessments.		

Completion date,
07/10/2016

Instructors,

Approval date of department

Director of department,