

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Babeș-Bolyai
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Departamentul de Informatică
1.4 Domeniul de studii	Informatică
1.5 Ciclul de studii	Licență
1.6 Programul de studiu / Calificarea	Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Securitate Software						
2.2 Titularul activităților de curs	Conf. Dr. Mihai SUCIU						
2.3 Titularul activităților de seminar	Conf. Dr. Mihai SUCIU						
2.4 Anul de studiu	3	2.5 Semestrul	5	2.6. Tipul de evaluare	C	2.7 Regimul disciplinei	Optională
2.8 Codul disciplinei	MLR8114						

3. Timpul total estimat (ore pe semestru al activităților didactice)

3.1 Număr de ore pe săptămână	4	Din care: 3.2 curs	2	3.3 seminar/laborator	0+1+1
3.4 Total ore din planul de învățământ	56	Din care: 3.5 curs	28	3.6 seminar/laborator	14
Distribuția fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie și notițe					15
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					10
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					17
Tutoriat					0
Examinări					2
Alte activități:					0
3.7 Total ore studiu individual	44				
3.8 Total ore pe semestru	100				
3.9 Numărul de credite	4				

4. Precondiții

4.1 de curriculum	<input type="checkbox"/> Arhitectura sistemelor de calcul <input type="checkbox"/> Sisteme de operare <input type="checkbox"/> Structuri de date și algoritmi <input type="checkbox"/> Baze de date <input type="checkbox"/> Programare web
4.2 de competențe	Programare în C, cunoștințe de baza ale arhitecturii Intel x86, elemente de bază în programarea web și SQL.

5. Condiții

5.1 De desfășurare a cursului	<input type="checkbox"/> Sală dotată cu videoproiector
5.2 De desfășurare a seminarului/laboratorului	<input type="checkbox"/> Laborator cu calculatoare cu Visual Studio (C++)

6. Competențele specifice acumulate

Competențe profesionale	<ul style="list-style-type: none"><input type="checkbox"/> C6.1 Identificarea conceptelor și modelelor de baza pentru sisteme de calcul și rețele de calculatoare.<input type="checkbox"/> C6.2 Identificarea și explicarea arhitecturilor de bază pentru organizarea și gestiunea sistemelor și a rețelelor.<input type="checkbox"/> C6.4 Efectuarea de măsurători de performanță pentru timpi de răspuns, consum de resurse; stabilirea drepturilor de acces.
Competențe transversale	<ul style="list-style-type: none"><input type="checkbox"/> CT1 Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor și a normelor de etică profesională<input type="checkbox"/> CT3 Utilizarea unor metode și tehnici eficiente de învățare, informare, cercetare și dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice și de comunicare în limba română și într-o limbă de circulație internațională

7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

7.1 Obiectivul general al disciplinei	Capacitatea evaluării caracteristicilor de securitate ale unei aplicații software la nivelul codului sursă. Dobândirea deprinderilor fundamentale minimale de scriere a unui cod sursă fără vulnerabilități.
7.2 Obiectivele specifice	<ul style="list-style-type: none"><input type="checkbox"/> Cunoașterea mecanismelor de bază ce definesc securitatea sistemului și a mediului software în care se execută o aplicație (i.e. modelul de securitate), cum ar fi: permisiunile de acces, politicile de securitate, interacțiunea cu mediul exterior etc.<input type="checkbox"/> Cunoașterea principalelor tipuri de vulnerabilități software, precum: utilizarea datelor utilizator nevalidate corespunzător, interacțiunea necontrolată directă sau indirectă cu mediul exterior aplicației etc.<input type="checkbox"/> Deprinderea unor tehnici eficiente de studiere și evaluare a unui cod sursă din perspectiva securității și capacitatea de a identifica posibile vulnerabilități.<input type="checkbox"/> Capacitatea de a evalua implicațiile unei vulnerabilități descoperite.<input type="checkbox"/> Cunoașterea tehnicilor și a bibliotecilor de funcții utile în scrierea unui cod sursă fără vulnerabilități și capacitatea de a le utiliza în situații reale.

8. Conținuturi

8.1 Curs		Metode de predare	Observații
1	Concepte și aspecte de bază referitoare la vulnerabilitățile software și la metodele și uneltele de dezvoltare a unui software fără vulnerabilități și de evaluare a unui software din perspectiva posibilelor vulnerabilități	Expunere la tablă, prezență cu video proiectorul, discuții, probleme scurte	
2	Vulnerabilități de corupere a memoriei (<i>buffer/integer overflow etc.</i>)		
3	Vulnerabilități specifice limbajului C: limite aritmetice (de reprezentare), conversii de tip, pointeri etc.		
4	Vulnerabilități în componentele structurale ale unei aplicații software (<i>Program building blocks</i>)		
5	Vulnerabilități în utilizarea și manipularea șirurilor de caractere și metacaractere		
6	Vulnerabilități specifice sistemelor de operare UNIX		
7	Vulnerabilități specifice sistemelor de operare Windows		
8	Vulnerabilități de sincronizare (în situații de concurență)		
9	Vulnerabilități Web: injectare cod SQL, XSS, XSRF etc.		
10	Vulnerabilități de criptografie: parole vulnerabile, numere aleatoare previzibile etc.		
11	Vulnerabilități specifice codului aplicațiilor ce folosesc comunicarea în rețelele de calculatoare		
12	Metode de proiectare corectă a aplicațiilor din perspectiva securității: principii de proiectare, definirea modelului de riscuri (<i>threat modeling</i>), evaluare design etc.		
13	Metode de implementare corectă a unei aplicații software din perspectiva securității: metode și modele de dezvoltare a aplicațiilor (<i>Waterfall, agile</i>), cele mai frecvente și mai periculoase riscuri și vulnerabilități, tehnici de defensive de scriere a codului (<i>defensive coding techniques</i>)		
14	Metode de evaluare a (codului) unei aplicații din perspectiva securității: asigurarea calității, testare, gestiunea vulnerabilităților identificate		
8.2 Seminar / laborator		Metode de predare	Observații
1	Unelte utile în identificarea și evaluarea vulnerabilităților unui cod sursă: navigatoare prin codul sursă, depanatoare, navigatoare prin codul executabil (binar), testare <i>fuzzy</i>	Scurte expuneri la tablă, tutoriale, ghiduri de lucru, demonstrații <i>live</i> , explicații suplimentare, discuții, propunerea spre rezolvare a unor probleme de diferite tipuri și grade de complexitate.	
2	Tehnici de evitare, detecție și evaluare a vulnerabilităților de corupere a memoriei și specifice limbajului C		
3	Tehnici de evitare, detecție și evaluare a vulnerabilităților de utilizare și gestionare a șirurilor de caractere și metacaractere		
4	Tehnici de evitare, detecție și evaluare a vulnerabilităților specifice sistemului de operare Linux		
5	Tehnici de evitare, detecție și evaluare a vulnerabilităților sistemelor de operare Windows		
6	Tehnici de evitare, detecție și evaluare a vulnerabilităților de sincronizare		
7	Tehnici de evitare, detecție și evaluare a vulnerabilităților aplicațiilor Web și aplicațiilor de rețea		
Bibliografie			
1. M. Down, J. McDonald, J. Schuh, „ <i>The Art of Software Security Assessment. Identifying and Preventing</i>			

Software Vulnerabilities”, AddisonWesley, 2007

2. M. Howard, D. LeBlanc, J. Viega, „*24 Deadly Sins of Software Security. Programming Flows and How to Fix Them*”, McGraw Hill, 2010

3. M. Howard, D. LeBlanc, „*Writing Secure Code for Windows Vista*”, Microsoft Press, 2007

4. G. McGraw, „*Software Security: Building Security In*”, AddisonWesley, 2006

5. R. Seacord, „*CERT C Coding Standard: 98 Rules for Developing Safe, Reliable, and Secure Systems*”, AddisonWesley, 2nd edition, 2014

6. „*Common Weaknesses Enumeration (WCE)*”, online: <http://cwe.mitre.org/data/index.html>

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului

Se realizează prin discuții periodice cu reprezentanți ai angajatorilor semnificativi din domeniul securității informației.

Cursuri referitoare la aspecte de securitate în dezvoltarea aplicațiilor și domenii adiacente (de exemplu teste de penetrare) sunt prezente în cadrul multor alte masterate din domeniul securității calculatoarelor și a informațiilor, la universități din țară și străinătate, cum ar fi:

· *Securitatea sistemelor software*, Master de Securitatea informației, Universitatea Al. I. Cuza, Iași, Facultatea de calculatoare, <http://profs.info.uaic.ro/~webdata/planuri/master/MISS1FS03.pdf>

· *Securitatea sistemelor și aplicațiilor*, Master de Securitatea tehnologiei informației, Academia Tehnică Militară, București, <http://mta.ro/masterat/masterinfosec/curricula2013.html>

· *Secure Software Systems*, Master of Science in Information Security, Carnegie Mellon University, SUA, <http://www.ini.cmu.edu/degrees/msis/courses.html>

· *Software Security*, Master in Information Security, Royal Holloway University of London, Information Security Group,

[https://www.royalholloway.ac.uk/isg/documents/pdf/coursespecs\(msc\)/modules201314/iy5607softwa_resecurityspec1314.pdf](https://www.royalholloway.ac.uk/isg/documents/pdf/coursespecs(msc)/modules201314/iy5607softwa_resecurityspec1314.pdf)

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 metode de evaluare	10.3 Pondere din nota finală
10.5 Seminar/laborator	Abilitatea de rezolvare a unor probleme specifice domeniului · Prezență, (inter)activitate în timpul orelor de laborator/proiect.	Realizarea activităților de laborator și/sau rezolvarea temelor de casă și/sau a unor probleme în cadrul unui test practic	40%(activitate pe parcurs)+40%(test grilă colocviu)+20%(proiect laborator)
10.6 Standard minim de performanță			
<input type="checkbox"/> Capacitatea de a defini vulnerabilitățile software fundamentale, precum: bufferoverflow, injectare code SQL, XSS etc.			
<input type="checkbox"/> Capacitatea de a identifica vulnerabilitățile software fundamentale și de a corecta codul (demonstrate în cadrul exercițiilor de laborator și a evaluării finale).			
<input type="checkbox"/> Nota finala:			

- 40% activitate pe parcurs, minim nota 5 la fiecare activitate notata
- 20% proiect laborator
- 40% examen colocviu, minim nota 5 pentru promovare
- Prezenta la activitățile de laborator
- Studenții vor fi consecvenți și vor participa în același interval orar la activitățile de seminar și laborator.
- Activitățile de pe parcursul semestrului nu se pot repeta în sesiunea normală sau sesiunea de restanțe.

Data completării

28.04.2023

Titular de curs

Conf. Dr. Mihai SUCIU

Titular de seminar / laborator / proiect

Conf. Dr. Mihai SUCIU

Data avizării în departament

.....

Director de departament

Prof. Dr. Laura DIOȘAN