FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituţia de învăţământ superior	Universitatea Tehnică din Cluj-Napoca
1.2 Facultatea	Automatică și Calculatoare
1.3 Departamentul	Calculatoare
1.4 Domeniul de studii	Calculatoare si Tehnologia Informatiei
1.5 Ciclul de studii	Master
1.6 Programul de studii / Calificarea	Securitatea Informatiilor si Sistemelor de calcul/ Master
1.7 Forma de învăţământ	IF – învăţământ cu frecvenţă
1.8 Codul disciplinei	1.

2. Date despre disciplină

2.1 Denumirea disciplinei				***Probleme de securitate la nivel de cod sursă***	
2.2 Titularii de curs				Conf. dr.ing. Adrian COLEȘA - (adrian.colesa@cs.utcluj.ro)	
2.3 Titularul/Titularii activităţilor de seminar/laborator/proiect				Răzvan Rațiu - razvan.ratiu95@gmail.com Laszlo Ciople - laszlo.ciople@gmail.com	
2.4 Anul de studiu	I	2.5 Semestrul	1	2.6 Tipul de evaluare *(E – examen, C – colocviu, V – verificare)*	E
2.7 Regimul disciplinei	*DA – de aprofundare, DS – de sinteza, DC – complementară*				DS
	DI – Impusă, DOp – opțională, DFac – facultativă				DI

3. Timpul total estimat

3.1 Număr de ore pe săptămână	3	din care:	Curs	2	Seminar		Laborator	1	Proiect	
3.2 Număr de ore pe semestru	42	din care:	Curs	28	Seminar		Laborator	14	Proiect	

3.3 Distribuţia fondului de timp (ore pe semestru) pentru:	
(a) Studiul după manual, suport de curs, bibliografie şi notiţe	18
(b) Documentare suplimentară în bibliotecă, pe platforme electronice de specialitate şi pe teren	18
(c) Pregătire seminarii / laboratoare, teme, referate, portofolii şi eseuri	45
(d) Tutoriat	0
(e) Examinări	2
(f) Alte activităţi:	0
3.4 Total ore studiu individual (suma (3.3(a)…3.3(f)))	83
3.5 Total ore pe semestru (3.2+3.4)	125
3.6 Numărul de credite	5

4. Precondiţii (acolo unde este cazul)

4.1 de curriculum	Programarea calculatoarelor, Structuri de date și algoritmi, Sisteme de operare
4.2 de competenţe	Programare în C, cunoștințe de bază ale arhitecturii Intel x86, elemente de bază în programarea web

5. Condiţii (acolo unde este cazul)

5.1. de desfăşurare a cursului	Tabla, proiector, calculatoare
5.2. de desfăşurare a seminarului / laboratorului / proiectului	Tabla, proiector, calculatoare

6. Competenţele specifice acumulate

6.1 Competenţe profesionale	C1 - Identificarea și înțelegerea problemelor de securitate ce pot apărea în diverse domenii de utilizare a sistemelor de calcul. Aplicarea adecvată a elementelor de bază ale managementului securității și ale modalităților de evaluare și management al riscurilor de securitate informatice • C1.1 – Cunoașterea noțiunilor, conceptelor și principiilor teoretice și practice avansate aferente domeniului general al securității informațiilor și sistemelor de calcul. Cunoașterea conceptelor aferente riscurilor, evaluării riscurilor și managementului securității • C1.2 – Înțelegerea riscurilor de securitate specifice unor situații noi și

	legătura lor cu cele cunoscute anterior. Prevederea scenariilor posibile de securitate, în momentul în care soluțiile de securitate cunoscute sunt folosite într-o situație sau domeniu nou C4 - Proiectarea și dezvoltarea de software cu un înalt grad de securitate, de soluții și unelte de securitate • C4.1 – Cunoașterea principiilor și noțiunilor de bază necesare dezvoltării și testării unui cod sigur din punctul de vedere al securității. Cunoașterea claselor uzuale de software și unelte de securitate. Cunoașterea arhitecturilor de SO și platformelor necesare dezvoltării soluțiilor de securitate • C4.3 – Dezvoltarea unor module software complexe respectând principiile metodologiilor de dezvoltare corectă a unui software din perspectiva securității. Dezvoltarea unor utilitare de analiză sau de validare a securității • C4.4 – Evaluarea unor proiecte software existente și identificarea greșelilor de securitate din punctul de vedere al arhitecturii, modului de programare sau procedurilor de testare. Propunerea unor noi metode de dezvoltare și testare • C4.5 – Dezvoltarea unor module software sau a unor utilitare care să ajute la asigurarea unui înalt grad de securitate. Propunerea unor scenarii și modalități de testare a unor proiecte existente, cu scopul de a verifica și asigura calitatea lor din punctul de vedere al securității
6.2 Competenţe transversale	N/A

7. Obiectivele disciplinei

7.1 Obiectivul general al disciplinei	Capacitatea evaluării caracteristicilor de securitate ale unei aplicații software la nivelul codului ei sursă. Dobândirea deprinderilor fundamentale de scriere a unui cod sursă fără vulnerabilități.
7.2 Obiectivele specifice	1. 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. 2. Cunoașterea principalelor tipuri de vulnerabilități software, precum: utilizarea datelor utilizator nevalidate corespunzător, interacțiunea necontrolată directă sau indirectă cu mediul exterior aplicației etc. 3. Deprinderea unor tehnici eficiente de studiere și evaluare a unui cod sursă din perspectiva securității și capacitatea de a identifica posibile vulnerabilități. 4. Capacitatea de a evalua implicațiile unei vulnerabilități descoperite. 5. 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. Conţinuturi

8.1 Curs	Nr.ore	Metode de predare	Observații
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	2	Expunere la tablă, prezentare cu video-proiectorul, discuții, probleme scurte	
Vulnerabilități de corupere a memoriei (*buffer/integer overflow etc.*)	2		
Vulnerabilități specifice limbajului C: limite aritmetice (de reprezentare), conversii de tip, pointeri etc.	2		
Vulnerabilități în componentele structurale ale unei aplicații software (*Program building blocks*)	2		
Vulnerabilități în utilizarea și manipularea șirurilor de caractere și meta-caractere	2		

Vulnerabilități specifice sistemelor de operare UNIX	2		
Vulnerabilități specifice sistemelor de operare Windows	2		
Vulnerabilități de sincronizare (în situații de concurență)	2		
Vulnerabilități Web: injectare cod SQL, XSS, XSRF etc.	2		
Vulnerabilități de criptografie: parole vulnerabile, numere aleatoare previzibile etc.	2		
Vulnerabilități specifice codului aplicațiilor ce folosesc comunicarea în rețelele de calculatoare	2		
Metode de proiectare corectă a aplicațiilor din perspectiva securității: principii de proiectare, definirea modelului de riscuri (threat modeling), evaluare design etc.	2		
Metode de implementare corectă a unei aplicații software din perspectiva securității: metode și modele de dezvoltare a aplicațiilor (Waterfall, Agile), cele mai frecvente și mai periculoase riscuri și vulnerabilități, tehnici de defensive de scriere a codului (*defensive coding techniques*)	2		
Metode de evaluare a (codului) unei aplicații din perspectiva securității: asigurarea calității, testare, gestiunea vulnerabilităților identificate	2		

Bibliografie *(bibliografia minimală a disciplinei conținând cel puțin o lucrare bibliografică de referință a disciplinei, care există la dispoziția studenților într-un număr de exemplare corespunzător)*

1. M. Down, J. McDonald, J. Schuh, „*The Art of Software Security Assessment. Identifying and Preventing Software Vulnerabilities*”, Addison-Wesley, 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*”, Addison-Wesley, 2006
5. R. Seacord, „CERT C Coding Standard: 98 Rules for Developing Safe, Reliable, and Secure Systems”, Addison-Wesley, 2nd edition, 2014
6. -, „*Common Weaknesses Enumeration* (WCE)”, on-line: http://cwe.mitre.org/data/index.html

8.2 Aplicații (seminar/laborator/proiect)*	Nr.ore	Metode de predare	Observații
Unelte utile în identificarea și evaluarea vulnerabilităților unui cod sursă: navigatoare prin codul sursă, depanatoare, navigatoare prin codul executabil (binar), testare *fuzzy*	1	Scurte expuneri la tablă, tutoriale, ghiduri de lucru, demonstrații *live*, explicații suplimentare, discuții, propunerea spre rezolvare a unor probleme de diferite tipuri și grade de complexitate	
Tehnici de evitare, detecție și evaluare a vulnerabilităților de corupere a memoriei si specifice limbajului C	1		
Tehnici de evitare, detecție și evaluare a vulnerabilităților de utilizare și gestionare a șirurilor de caractere și meta-caractere	1		
Tehnici de evitare, detecție și evaluare a vulnerabilităților specifice sistemului de operare Linux	1		
Tehnici de evitare, detecție și evaluare a vulnerabilităților sistemelor de operare Windows	1		
Tehnici de evitare, detecție și evaluare a vulnerabilităților de sincronizare	1		
Tehnici de evitare, detecție și evaluare a vulnerabilităților aplicațiilor Web si aplicațiilor de rețea	1		

Bibliografie *(bibliografia minimală pentru aplicații conținând cel puțin o lucrare bibliografică de referință a disciplinei care există la dispoziția studenților într-un număr de exemplare corespunzător)*

1. M. Down, J. McDonald, J. Schuh, „*The Art of Software Security Assessment. Identifying and Preventing Software Vulnerabilities*”, Addison-Wesley, 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*”, Addison-Wesley, 2006
5. R. Seacord, „CERT C Coding Standard: 98 Rules for Developing Safe, Reliable, and Secure Systems”, Addison-Wesley, 2nd edition, 2014
6. -, „*Common Weaknesses Enumeration* (WCE)”, on-line: http://cwe.mitre.org/data/index.html

9. Coroborarea conţinuturilor disciplinei cu aşteptările reprezentanţilor comunităţii epistemice, asociaţiilor profesionale şi angajatorilor reprezentativi din domeniul aferent programului

Se realizează prin discuții periodice cu reprezentanți ai angajatorilor importanți 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/curricula2014.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/iy5607softwaresecurityspec1314.pdf

10. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Curs	Abilitatea de definire a conceptelor specifice problemelor de securitate la nivel de cod sursă și de expunere a metodelor de evaluare și dezvoltare corectă a unui cod sursă din perspectiva securității. Abilitatea de rezolvare a unor probleme specifice domeniului. Prezență, (inter)activitate în timpul orelor de curs.	Examen scris și/sau tip grilă (pe calculator, utilizând platforma Moodle) și/sau prezentarea unei teme de cercetare din domeniul cursului In situații excepționale, care necesită desfășurarea activităților didactice de la distanță, examinarea poate avea loc online, de la distanță, folosind platformele Moodle și Teams.	50%
Seminar	-	-	-
Laborator	Abilitatea de rezolvare a unor probleme specifice domeniului. Prezență, (inter)activitate în timpul orelor de laborator.	Realizarea activităților de laborator și rezolvarea temelor de casă și/sau a unor probleme în cadrul unui examen practic. In situații excepționale, care necesită desfășurarea activităților didactice de la distanță, examinarea poate avea loc online, de la distanță, folosind platformele Moodle și Teams.	50%
Proiect	-	-	-

Standard minim de performanţă

***Curs*.** Prezența la **minim 50%** din orele de curs, pentru admiterea la examenul final. Capacitatea de a defini vulnerabilitățile software fundamentale, precum: buffer-overflow, injectare code SQL, XSS etc.

***Aplicații*.** Prezența la laborator **obligatorie 100%** (2 laboratoare se pot recupera în timpul semestrului, iar alte 2 în sesiunile de restanțe) pentru admiterea la examenul final. Capacitatea de a identifica vulnerabilitățile software fundamentale și de a corecta codul (demonstrate în cadrul exercițiilor de laborator și a examenului final).

Data completării:	Titulari	Titlu Prenume NUME	Semnătura
	Curs	Conf. dr.ing. Adrian COLEȘA	
	Aplicații	Ing. Răzvan Rațiu Ing. LAszlo Ciople	

Data avizării în Consiliul Departamentului Calculatoare	Director Departament Prof.dr.ing. Rodica Potolea
Data aprobării în Consiliul Facultății de Automatică și Calculatoare	Decan Prof.dr.ing. Liviu Miclea