

## A TANTÁRGY ADATLAPJA

### 1. A képzési program adatai

1.1 Felsőoktatási intézmény	Babeş-Bolyai Tudományegyetem
1.2 Kar	Matematika és Informatika
1.3 Intézet	Magyar Matematika és Informatika
1.4 Szakterület	Informatika
1.5 Képzési szint	Alap
1.6 Szak / Képesítés	Informatika

### 2. A tantárgy adatai

2.1 A tantárgy neve	<b>Szoftverbiztonság</b> Securitate software – Secure coding						
A tantárgy kódja:	<b>MLM5086</b>						
2.2 Az előadásért felelős tanár neve	Dr. Robu Judit egyetemi docens						
2.3 A szemináriumért felelős tanár neve	Fülöp Botond (Bitdefender)						
2.4 Tanulmányi év	3	2.5 Félév	5	2.6 Értékelés módja	kollokvium	2.7 Tantárgy típusa	választható szaktárgy

### 3. Teljes becsült idő (az oktatási tevékenység féléves óraszámja)

3.1 Heti óraszám	4	melyből: 3.2 előadás	2	3.3 labor/projekt	1+1
3.4 Tantervben szereplő össz-óraszám	56	melyből: 3.5 előadás	28	3.6 labor/projekt	28
A tanulmányi idő elosztása:					óra
A tankönyv, a jegyzet, a szakirodalom vagy saját jegyzetek tanulmányozása					13
Könyvtárban, elektronikus adatbázisokban vagy terepen való további tájékozódás					8
Szemináriumok / laborok, házi feladatok, portfóliók, referátumok, esszék kidolgozása					23
Egyéni készségfejlesztés (tutorálás)					0
Vizsgák					0
Más tevékenységek: .....					
3.7 Egyéni munka össz-óraszámja	44				
3.8 A félév össz-óraszámja	100				
3.9 Kreditszám	4				

### 4. Előfeltételek (ha vannak)

4.1 Tantervi	<ul style="list-style-type: none"> <li>• Számítógép architektúra</li> <li>• Operációs rendszerek</li> <li>• Adatszerkezetek</li> <li>• Adatbázisok</li> <li>• Webprogramozás</li> </ul>
4.2 Kompetenciabeli	<ul style="list-style-type: none"> <li>• C programozási készség, x86 architektúra ismerete, alap webprogramozás és SQL ismeretek</li> </ul>

### 5. Feltételek (ha vannak)

5.1 Az előadás	<ul style="list-style-type: none"> <li>• Táblával és videoprojektossal felszerelt előadó</li> </ul>
----------------	---

lebonyolításának feltételei	
5.2 A szeminárium / labor / projekt lebonyolításának feltételei	<ul style="list-style-type: none"> <li>• Számítógépes terem</li> </ul>

## 6. Elsajátítandó jellemző kompetenciák

<b>Szakmai kompetenciák</b>	<p><b>C1.1</b> A programozási paradigmák és a specifikus nyelvi mechanizmusok megfelelő leírása, valamint a szemantikai és a szintaktikai vonatkozások közötti különbség meghatározása</p> <p><b>C1.2</b> A meglévő szoftveralkalmazások magyarázata absztrakciós szintek szerint (architektúra, csomagok, osztályok, metódusok) az alapismeretek használatával</p> <p><b>C1.4</b> Alkalmazások tesztelése adott tesztelési terv alapján.</p> <p><b>C2.1</b> A szoftverrendszerek megfelelő fejlesztési módszereinek beazonosítása.</p> <p><b>C2.4</b> Megfelelő kritériumok és módszerek használata az alkalmazások értékeléséhez.</p> <p><b>C6.1</b> Számítási rendszerek és számítógépes hálózatok alapkoncepcióinak és modelljeinek azonosítása</p>
<b>Transzverzális kompetenciák</b>	<p><b>CT1</b> A szervezett és hatékony munka szabályainak, a didaktikai-tudományos területhez való felelősségteljes hozzáállás alkalmazása a saját potenciál kreatív értékesítéséhez, a szakmai etika alapelveinek és normáinak tiszteletben tartásával</p>

## 7. A tantárgy célkitűzései (az elsajátítandó jellemző kompetenciák alapján)

7.1 A tantárgy általános célkitűzése	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 A tantárgy sajátos célkitűzései	<ul style="list-style-type: none"> <li>• 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.</li> <li>• Cunoașterea principalelor tipuri de vulnerabilități software, precum: utilizarea datelor utilizator nevalidate corespunzător, interacțiunea necontrolată directă sau indirectă cu mediul exterior aplicației etc.</li> <li>• Deprinderea unor tehnici eficiente de studiere și evaluare a unui cod sursă din perspectiva securității și capacitatea de a identifica posibile vulnerabilități.</li> <li>• Capacitatea de a evalua implicațiile unei vulnerabilități descoperite.</li> <li>• Cunoașterea tehnicilor și a bibliotecilor de funcții utile în scrierea unui cod sursă fără vulnerabilități și capacitatea de a le utiliza în situații reale.</li> </ul>

## 8. A tantárgy tartalma

8.1 Előadás	Didaktikai módszerek	Megjegyzések
1. Szoftver biztonsági rések, biztonságos szoftverfejlesztés, biztonság kiértékelése.	Problémafelvetés, előadás, megbeszélés	
2. Memória sebezhetőség (puffer/integer túlsordulás, stb)		
3. C-specifikus sebezhetőségek: numerikus adatok ábrázolása, határok, típuskonverziók, mutatók, stb.		
4. Szoftver alkalmazások alkotóelemeihez kapcsolódó sebezhetőségek.		

5. Karaktorsorokkal és metakarakterekkel kapcsolatos sebezhetőségek.		
6. UNIX operációs rendszerekhez kapcsolódó sebezhetőségek.		
7. Windows operációs rendszerekhez kapcsolódó sebezhetőségek.		
8. Szinkronizálás, versenyhelyzet		
9. Webes alkalmazások sebezhetősége: SQL injection, XSS, XSRF, stb.		
10. Kriptográfiai sebezhetőségek: feltörhető jelszavak, megjósolható véletlen számok, stb.		
11. Hálózati kommunikációhoz kapcsolódó sebezhetőségek.		
12. Alkalmazások helyes tervezése, biztonsági megközelítés: alapelvek, fenyegetési modell, kiértékelés		
13. Alkalmazások biztonságkritikus implementálása (defensive coding techniques)		
14. Kód ellenőrzés, tesztelés, beazonosított sebezhetőségek kezelése		

#### Könyvészet

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 Flaws and How to Fix Them*, McGraw Hill, 2010
3. R. Anderson, *Security Engineering*, Wiley, <sup>2</sup>2008.
4. C. P. Pfleeger, S. L. Pfleeger, J. Margulies: *Security in Computing*, Prentice Hall, <sup>5</sup>2015.
5. M. Howard, D. LeBlanc, *Writing Secure Code for Windows Vista*, Microsoft Press, 2007
6. G. McGraw, *Software Security: Building Security In*, Addison-Wesley, 2006
7. R. Seacord, *CERT C Coding Standard: 98 Rules for Developing Safe, Reliable, and Secure Systems*, Addison-Wesley, <sup>2</sup>2014
8. -, *Common Weaknesses Enumeration(WCE)*, on-line: <http://cwe.mitre.org/data/index.html>

8.2 Szeminárium / Labor / Projekt	Didaktikai módszerek	Megjegyzések
1. Kódbeli sebezhetőségek beazonosítására használatos eszközök	Elméleti összefoglaló, sebezhetőségek gyakorlati bemutatása, egyéni munka	
2. Memória sebezhetőség		
3. A C nyelv sebezhetősége		
4. String-ek és metakarakterek		
5. Linux biztonsági rések		
6. Windows biztonsági rések		
7. Webes biztonsági rések		

#### 9. A tantárgy tartalmának összhangba hozása az episztemikus közösségek képviselői, a szakmai egyesületek és a szakterület reprezentatív munkáltatói elvárásaival.

- A tantárgy tartalma megegyezik az egyetemi oktatásban a fontosabb egyetemeken oktatott „Software Security” tárgy hagyományos tartalmával.

## 10. Értékelés

Tevékenység típusa	10.1 Értékelési kritériumok	10.2 Értékelési módszerek	10.3 Aránya a végső jegyben
10.4 Előadás	Alapfogalmak ismerete	Quiz	20 %
10.5 Szeminárium / Labor	Laborfeladatok,	Minden héten gyakorlati feladatok, a félév végén összefoglaló gyakorlati feladat: sebezhetőségek azonosítása és javítása 2 programban	80 %
10.6 A teljesítmény minimumkövetelményei			
<ul style="list-style-type: none"><li>• Laborfeladatok elkészítése, átmenő jegy elméleti és gyakorlati felmérőn.</li></ul>			

Kitöltés dátuma

2019.04.18.

Előadás felelőse

Dr. Robu Judit docens

Szeminárium felelőse

Dr. Robu Judit docens

Az intézeti jóváhagyás dátuma

2019.04.20.

Intézetigazgató,

Dr. András Szilárd docens