VARNOST PROGRAMOV

UČNI NAČRT PREDMETA/COURSE SYLLABUS

Predmet:	Varnost programov
Course title:	Software Security
Članica nosilka/UL Member:	UL FRI

Študijski programi in stopnja	Študijska smer	Letnik	Semestri	Izbirnost
Računalništvo in informatika, prva stopnja, visokošolski strokovni (v postopku)	Ni členitve (študijski program)		2. semester	izbirni

Univerzitetna koda predmeta/University course code:	0643448
Koda učne enote na članici/UL Member course code:	63774

Predavanja /Lectures	Seminar /Seminar	Vaje /Tutorials	Klinične vaje /Clinical tutorials	Druge oblike študija /Other forms of study	Samostojno delo /Individual student work	ECTS
45		30			105	6

Nosilec predmeta/Lecturer:	Matevž Pesek

Vrsta predmeta/Course type:	izbirni-strokovni/elective-vocational

Jeziki/Languages:		
	Predavanja/Lectures:	
	Vaje/Tutorial:	

Pogoji za vključitev v delo oz. za opravljanje študijskih obveznosti:	Prerequisites:
Vpis v letnik.	Enrollment in the study year.

Vsebina:	Content (Syllabus outline):
Predmet je namenjen študentom, ki želijo primarno nadgraditi programerske in administrativne sposobnosti. Zaradi povečane uporabe tehnologije v sistemih, ki obravnavajo občutljive podatke, učinkovitost in hitrost nista več edina pogoja, ko pride do izdelave programske opreme. Vse večjo pozornost je treba posvečati varnosti in zanesljivosti. Pri predmetu bomo študente seznanili s sodobnimi tehnikami zlorabe programske opreme, kot tudi z varnostnimi mehanizmi vgrajenimi v programsko opremo, z namenom preprečevanja ter zajezitve dosega zlorabe. Pri vajah bomo na primerih posameznih tematik sprva na praktičnem primeru ne-varnega sistema preizkusili vdore v sistem, nato pa prikazali praktično	The course is intended for students who primarily want to upgrade their programming and system administration skills. Due to the increased use of technology in systems that deal with sensitive data, efficiency and speed are no longer the only conditions when it comes to software development. More and more attention must be paid to security and reliability. In the course, students will be introduced to modern software abuse techniques, as well as with security mechanisms built into the software, with the aim of preventing and limiting the scope of abuse. During the lab sessions, we will first demonstrate how to exploit on a practical example of an insecure system, and then demonstrate the practical writing of robust code that eliminates the problems of the individual topic.

pisanje robustne kode, ki odpravlja probleme posamezne tematike. Vsebine predmeta obsegajo naslednje tematike: • Uporabniški vnosi in problematika (sanitizacija, napadi s kompresijo) • Vrivanje zlonamerne kode v komunikacijo z zalednim delom (SQL, komentarji, timing napadi, eksfiltracija podatkov, pisanje datotek) • Prelivanje skozi tipe (velika števila, nepredstavljiva števila, velikost nizov) • Napadi skozi XML/HTML (escaping, stored/cross-site scripting, server-side request forgery) • Vrivanje preko formatov (executable regex in format string, format mismatching) • Dnevniški zapisi (uporabnost, vrivanje, monitoring, vrivanje ukazov) in revizijske sledi • Zloraba kriptografskih standardov (HMAC, podpisovanje, kodiranje CBC/ECB/GCM, podpisovanje za dokazovanje obstoja) • Avtentikacijski algoritmi (JWT, openID, Auth0) • Deserializacija objektov (nevarnosti, JSON dump, user state) • Problematika odvisnih knjižnic (omejevanje na različico knjižnice, supply chain napadi) • Race condition (skozi niti/procese) • Defenzivno programiranje (Preverjanje napak, pričakovanje izjem, preverjanje nedefiniranih rezultatov funkcij, sanitizacija vhodnih podatkov, povratek v stabilno stanje) • Orodja za razhroščevanje, ponovljivost prevedene kode, CI/CD • Penetracijski testi • statična analiza kode, fuzzer, orodja za avtomatično preverjanje kvalitete kode	The contents of the course include the following topics: • User inputs and related issues (sanitization, compression attacks) • Injecting malicious code into communication with the backend (SQL, comments, timing attacks, data exfiltration, writing files) • Casting through types (large numbers, unrepresentable numbers, string size) • Attacks through XML/HTML (escaping, stored/cross-site scripting, server-side request forgery) • Format attacks (executable regex and format string, format mismatching) • Logs (usability, injection, monitoring, drilling commands) and revisions • Abuse of cryptographic standards (HMAC, signing, CBC/ECB/GCM encoding, proof-of-existence signing) • Authentication algorithms (JWT, openID, Auth0) • Deserialization of objects (dangers, JSON dump, user state) • The issue of dependent libraries (library version limitation, supply chain attacks) • Race condition (through threads/processes) • Defensive programming (Error checking, expecting exceptions, checking undefined function results, sanitizing input data, returning to stable state) • Debugging tools, reproducibility of compiled code, CI/CD • Penetration tests • Static code analysis, fuzzer, tools for automatic code quality check

Temeljna literatura in viri/Readings:

Wenliang Du - Computer Security: A Hands-on Approach (2017), ISBN: 978-981-126-329-3 **Gerardus Blokdyk - Software Security Vulnerability A Complete Guide** (2020), ISBN: 978-1-86-732146-0 **Mathias Payer - Software Security: Principles, Policies, and Protection.** (2021), Dostopno na: https://nebelwelt.net/SS3P/ **William Stallings, Lawrie Brown - Computer Security: Principles and Practice** (2014), ISBN: 978-0-13-377392-7

Cilji in kompetence:	**Objectives and competences:**
Cilji predmeta so: • študente seznaniti z metodologijo odkrivanja in izogibanja napak, • predstaviti tehnične postopke, prisotne pri zaznavanju ranljivosti v programski kodi, • študentom podati znanje za izvedbo pogostejših tipov napadov na ranljivo programsko kodo skozi uporabniški vnos, protokole in tipične pomanjkljivosti sodobne programske opreme.	The goals and core skills of the module are to: • acquaint students with the methodology of error avoidance detection, • present the technical procedures involved in the detection of vulnerabilities in software code, • teach students how to perform common types of attacks on vulnerable software code through user input, protocols, and typical flaws in modern software.

Predvideni študijski rezultati:	**Intended learning outcomes:**
Po uspešno opravljenem predmetu bodo študenti: - podrobno poznali osnovne principe ranljivosti programske kode, - poglobljeno razumeli delovanje sodobnih varnostnih mehanizmov, - znali identificirati ranljivosti v programski kodi - znali oceniti doseg in škodo, ki jo lahko povzroči določena ranljivost, - znali odpraviti tipične ranljivosti v sodobnem okolju.	After successful completion of the module, the participants will be able to: - understand the basic principles of program code vulnerability, - understand in detail the operation of modern security mechanisms, - comprehend the vulnerabilities in the programming code, - be able to assess the range and damage that can be caused by a certain vulnerability, - be able how to eliminate typical vulnerabilities in the modern environment.

Metode poučevanja in učenja:	**Learning and teaching methods:**
Predavanja, praktične vaje in demonstracije, projektni način dela pri seminarjih in vajah.	Lectures, lab work, home assignments, project work.

Načini ocenjevanja:	**Delež/Weight**	**Assessment:**
Način (pisni izpit, ustno izpraševanje, naloge, projekt)		Type (examination, oral, coursework,project)
Sprotno preverjanje (domače naloge, projektno delo)	50,00 %	Continuous (home assignments, project work)
Končno preverjanje (pisni izpit)	25,00 %	Final (written exam)
Končno preverjanje (ustni izpit)	25,00 %	Final (oral exam)
Ocene: 6-10 pozitivno, 5 negativno (v skladu s Statutom UL)		Scale: 6-10 pass, 5 and below fail (According to the rules and ordnances of the University of Ljubljana)

Reference nosilca/Lecturer's references:

Strokovne reference:

Dnevi slovenske informatike - okrogla miza: Kibernetska varnost države in državljanov v negotovih časih - Matevž Pesek, moderator; Boštjan Pavlin, Ministrstvo za obrambo; Uroš Svete, Urad Vlade RS za informacijsko varnost; Gorazd Božič, SI-CERT; Gregor Spagnolo, SSRD.IO - 11.5.2022, Portorož

Mentorstvo – European Cybersecurity Challenge. Dunaj, 13.9. - 16.9.2022

Juvan, Andraž, Pesek, Matevž. Vzpostavitev in vzdrževanje infrastrukture na kibernetskih tekmovanjih, v recenzijskem postopku – Uporabna informatika, april 2023

Znanstvene reference:

PESEK, Matevž, VUČKO, Žiga, ŠAVLI, Peter, KAVČIČ, Alenka, MAROLT, Matija. Troubadour: a gamified e-learning platform for ear training. IEEE access. May 2020, vol. 8, str. 97090-97102, ilustr. ISSN 2169-3536. https://ieeexplore.ieee.org/abstract/document/9093057, DOI: 10.1109/ACCESS.2020.2994389.

PESEK, Matevž, JUVAN, Andraž, JAKOŠ, Jure, KOŠMRLJ, Janez, MAROLT, Matija, GAZVODA, Martin. Database independent automated structure elucidation of organic molecules based on IR, 1H NMR, 13C NMR, and MS data. Journal of chemical information and modeling. 22 Feb. 2021, vol. 61, iss. 2, str. 756-763, ilustr. ISSN 1549-9596.

KAVČIČ, Alenka, PESEK, Matevž, BOHAK, Ciril, MAROLT, Matija. Introducing on-site customers in agile software development projects: an alternative approach to project work in engineering education. International journal of engineering education. 2018, no. 2, part a, str. 482-496, ilustr. ISSN 0949-149X.