Principali informazioni sull'insegnamento

Denominazione dell'insegnamento	**Progettazione di Sistemi Sicuri**
Corso di studio	Laurea Magistrale in Sicurezza Informatica
Anno Accademico	2024/25
Crediti formativi universitari (CFU) / European Credit Transfer and Accumulation System (ECTS)	6 CFU
Settore Scientifico Disciplinare	INF/01 - ING/INF-05
Lingua di erogazione	Italiano
Anno di corso	Secondo
Periodo di erogazione	1^ semestre, le date esatte sono riportate nel manifesto/regolamento
Obbligo di frequenza	No, ma la frequenza è fortemente raccomandata
Sito web del corso di studio	https://www.uniba.it/it/corsi/cdl-sicurezza-informatica-taranto/corso-di-laurea-in-sicurezza-informatica-sede-di-taranto

Docente/i

Nome e cognome	Giuseppina Andresini
Indirizzo mail	giuseppina.andresini@uniba.it
Telefono	+39 080 5442407
Sede	Dipartimento di Informatica, Via Orabona 4, 70125, Bari. V piano stanza 510
Sede virtuale	https://elearning.uniba.it/
Sito web del docente	https://kdde.di.uniba.it/people/giuseppina-andresini/
Ricevimento (giorni, orari e modalità, es. su appuntamento)	Martedì dalle 10 alle 12 (su appuntamento inviando una email a giuseppina.andresini@uniba.it)

Syllabus

Obiettivi formativi	Sviluppare competenze sulla capacità di individuare le vulnerabilità di un sistema software che possono essere sfruttate dagli attaccanti, e le possibili soluzioni per prevenire e mitigare gli attacchi software.
Prerequisiti	Conoscenze di base di programmazione e linguaggi di programmazione (linguaggio C)
Contenuti di insegnamento (Programma)	1) Introduzione: cosa si intende per sicurezza informatica, triade CIA, classificazione del software malevolo per metodo di propagazione e payload (4 ore) 2) Attacchi di basso livello, basati sulla memoria (buffer overflow) (4 ore) 3) Difese contro gli attacchi basati sulla memoria, es. stack canaries, ASLR, integrità del flusso di controllo (CFI) (4 ore) 4) Sicurezza Web: SQL injection, Cross-site scripting (XSS), Cross-site request forgery (CSRF) e Session hijacking e difese basate su validazione dell'input (8 ore) 5) Test di penetrazione: panoramica di obiettivi, tecniche e strumenti fondamentali (4 ora) 6) Modelli di progettazione sicura. Modelli di minaccia e principi di progettazione della sicurezza. Idee di organizzazione. Esempi positivi e negativi di progettazione di sistemi nel mondo reale. (3 ore) 7) Revisione statica e automatica del codice: analisi statica ed esecuzione simbolica. Taint analysis come strumento di analisi statica (3 ore) 8) Introduzione alle vulnerabilità dei sistemi informatici basati su intelligenza artificiale (2 ore)
Testi di riferimento	• **Jon Erickson, Hacking, 2nd Edition The Art of Exploitation. 2008, 488 pp., ISBN-13: 978-1-59327-144-2.** • **William Stallings, Sicurezza dei computer e delle reti, 2022, 512 pp. ISBN-13: 9788891915290** **Testi per approfondimenti** • **OWASP Foundation, OWASP Testing Guide. 2020 - Version 5.0 https://owasp.org/www-project-web-security-testing-guide/stable/** Gli studenti che lo desiderano possono ottenere i testi in prestito dalla Biblioteca. Può convenire verificarne la disponibilità mediante il Sistema Bibliotecario di Ateneo https://opac.uniba.it/easyweb/w8018/index.php? e contattare la biblioteca per concordare il prestito.
Note ai testi di riferimento	Nel corso delle lezioni il docente illustrerà i concetti con l'ausilio di slide che sintetizzano (e talvolta integrano) i contenuti del testo di riferimento. Le slide saranno rese disponibili al termine di ogni lezione sulla piattaforma online (v. sopra 'sede virtuale'). Sulla piattaforma online sono disponibili: ● slide di supporto utilizzate dal docente durante le lezioni; ● note del docente sul progetto richiesto per l'esame; ● esempi di risoluzioni di esercitazioni di tipo Capture the flag (CTF) su macchine vulnerabili reperibili al sito web https://www.vulnhub.com/ utili come esempio per il progetto da portare all'esame.

	Note sul contenuto del progetto saranno fornite durante le lezioni (inoltre sulla piattaforma di elearning alla voce Progetto esame saranno fornite indicazioni utili per lo svolgimento del progetto).

Organizzazione della didattica

Ore

Totali	Didattica frontale	Progetto	Studio individuale
150 ore	32 ore	50 ore	68 ore

CFU/ETCS

6 CFU	4 CFU	2 CFU	

Metodi didattici

	Didattica in aula con lezioni di teoria.

Risultati di apprendimento previsti

Conoscenza e capacità di comprensione	Lo studente acquisterà la capacità di comprendere le criticità di sicurezza coinvolte nella progettazione di sistemi software, assumendo due prospettive: quella dell'utente malizioso, interessato ad attaccare i sistemi informatici e, specularmente, quella dello sviluppatore consapevole, interessato a mitigare le minacce di sicurezza esistenti nei sistemi durante la fase di sviluppo.
Conoscenza e capacità di comprensione applicate	Lo studente sarà in grado di utilizzare la conoscenza acquisita per individuare le vulnerabilità di un sistema anche attraverso analisi statica del codice e adottare le soluzioni software (es. scrittura di codice più robusto, per prevenire la possibilità di attacco)
Competenze trasversali	**Autonomia di giudizio** ○ Lo studente acquisterà la capacità di valutare in maniera autonoma le vulnerabilità di un sistema software e come applicare le strategie in grado di prevenire, mitigare i possibili attacchi

	Abilità comunicative ○ Lo studente sarà in grado di descrivere le vulnerabilità di un sistema software e le scelte intraprese per mitigare le stesse. **Capacità di apprendere in modo autonomo** ○ Lo studente acquisterà la capacità di apprendere le criticità nella progettazione di un sistema software assumendo due prospettive: quella dell'utente malizioso, interessato ad attaccare i sistemi stessi e, specularmente, quella dello sviluppatore consapevole, interessato a mitigare le minacce di sicurezza esistenti nei sistemi durante la fase di sviluppo

Valutazione	
Modalità di verifica dell'apprendimento	Prova scritta + Progetto · Prova scritta in aula, 9 domande (5 domande a risposta multipla e 4 domande a risposta aperta) su teoria in merito ad argomenti del sillabo; tempo assegnato 90 minuti; votazione massima 33/33. La prova scritta si ritiene superata se lo studente consegue una votazione maggiore uguale di 18/33. · Progetto: la consegna deve avvenire con almeno una settimana prima la data dell'appello tramite email istituzionale del docente di riferimento. Il progetto consiste in una relazione (in PDF o Word) sulla risoluzione di una esercitazione di penetration testing di tipo Capture the flag (CTF) su una macchina vulnerabile scelta dallo studente e concordata con il docente (esempio da vulnhub.com) che spieghi le scelte fatte dallo studente per risolvere la challenge (corredata da opportuni screen del proprio pc), le vulnerabilità riscontrate (coerentemente con i contenuti visti a lezione) e le tecniche difensive potrebbero essere adottate per rendere sicuro il sistema. La CTF scelta per il progetto dovrà essere concordata preventivamente con il docente tramite email istituzionale prima della consegna. Il progetto sarà valutato dal docente e discusso con lo studente durante l'appello. Il progetto assegnato è valido solo per gli appelli erogati nell'AA 2024-25. Il progetto si ritiene superato se lo studente consegue una votazione maggiore uguale di 18/33 all'atto della sua discussione.
Criteri di valutazione	● **Conoscenza e capacità di comprensione:** ○ Capacità di individuare le criticità di un sistema software e individuare le soluzioni in grado di prevenire e mitigare i possibili attacchi. ● **Conoscenza e capacità di comprensione applicate**: ○ Capacità di descrivere, con esempi pratici del mondo reale, le varie tecniche di attacco e di difesa adottate in sistemi software.

	• **Autonomia di giudizio**:
	  ○ Capacità di decidere quali strategie adottare in fase di progettazione dei sistemi software.
	• **Abilità comunicative:**
	  ○ Capacità di descrivere, attraverso gli argomenti trattati nel corso, le scelte intraprese per rafforzare la sicurezza dei sistemi software, durante la fase di sviluppo.
	• **Capacità di apprendere:**
	  ○ Comprensione dei concetti appresi durante il corso non solo applicati ai casi specifici descritti durante il corso ma applicabili anche a contesti differenti (esempio, sistemi software implementati in linguaggi di programmazione diversi o sistemi software basati su Intelligenza Artificiale).

Criteri di misurazione dell'apprendimento e di attribuzione del voto finale	Il voto finale è determinato con la seguente formula 1/2*(votazione prova scritta) + 1/2*(votazione del progetto).
Altro	Si suggerisce agli studenti di affidarsi esclusivamente alle informazioni/comunicazioni fornite sui siti ufficiali del Dipartimento di Informatica, ovvero sui gruppi social solo se costituiti e amministrati esclusivamente dai docenti dei relativi insegnamenti: • https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-di-laurea • https://www.uniba.it/it/ricerca/dipartimenti/informatica • https://elearning.uniba.it/ I programmi di tutti gli insegnamenti sono disponibili al seguente link: • https://elearning.uniba.it/ Le informazioni che tutti gli studenti dovrebbero conoscere sono scritte nei regolamenti didattici dei Corsi di Studi disponibili nel sito: • https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-di-laurea Si suggerisce agli studenti di diffidare delle informazioni e dei materiali circolanti su siti o gruppi social non ufficiali, poiché spesso sono risultati non affidabili, non corretti o incompleti. Per ogni dubbio, chiedere un incontro al docente secondo le modalità previste per il ricevimento. _____ Gli studenti potranno unirsi al forum del corso A.A. 2024/25 iscrivendosi al corso sulla piattaforma e-learning: https://elearning.uniba.it/

Main information on the course		
Course name	**Design of safe systems**	
Degree	Master degree in Computer Security	
Academic year	2024/2025	
European Credit Transfer and Accumulation System (ECTS), in Italian Crediti Formativi Universitari (CFU)		6 CFU (each CFU corresponds to 25 hours (h) of student's time); CFU are of type T1, T2 or T3 T1 = 8 h lecture + 17 h individual study T2 = 15 h practice + 10 h individual study T3 = 25 h individual study
Settore Scientifico Disciplinare	INF/01 - ING/INF-05	
Course language	Italian	
Course year	Second	
Course period	1^ semester, the exact dates are shown in the poster/regulations	
Course attendance requirement	None, but it is highly recommended to attend classes.	
Website of the Degree	https://www.uniba.it/it/corsi/cdl-sicurezza-informatica-taranto/corso-di-laurea-in-sicurezza-informatica-sede-di-taranto	

Teacher(s)	
Name and Surname	Giuseppina Andresini
email	giuseppina.andresini@uniba.it
phone	+39 080 5442407
office	Dipartimento di Informatica, Via Orabona 4, 70125, Bari. V piano stanza 510
e-learning platform	https://elearning.uniba.it/
Teacher's homepage	https://kdde.di.uniba.it/people/giuseppina-andresini/
Office hours	By appointment

Syllabus	
Course goals	Develop the ability to identify vulnerabilities in a software system that attackers can exploit and possible solutions to prevent and mitigate software attacks.
Prerequisites/requirements	*Basic knowledge of computer programming languages (in particular C)*
Course program	1) Introduction to computer security, CIA triad, classification of malicious software by propagation method and payload (4 hours) 2) Low-level, memory-based attacks (buffer overflows) (4 hours.) 3) Defenses strategies against memory-based attacks (e.g., stack canaries, ASLR, control flow integrity) (4 hours) 4) Web security: SQL injection, cross-site scripting (XSS), cross-site request forgery (CSRF) and Session hijacking and defenses based on input validation (8 hours) 5) Penetration testing: an overview of objectives, techniques, and tools (4 hours)

	6) Secure design patterns. Threat models and security design principles. Examples of real-world system design. (3 hours) 7) Static and automatic code review: static analysis and symbolic execution. Taint analysis (3 hours) 8) Introduction to vulnerabilities of artificial intelligence-based computer systems (2 hours)
Books of reference	• **Jon Erickson, Hacking, 2nd Edition The Art of Exploitation. 2008, 488 pp., ISBN-13: 978-1-59327-144-2.** • **William Stallings, Sicurezza dei computer e delle reti, 2022, 512 pp. ISBN-13: 9788891915290** **Books for further information** • **OWASP Foundation, OWASP Testing Guide. 2020 - Version 5.0 https://owasp.org/www-project-web-security-testing-guide/stable/** Students who wish can borrow texts from the Library. Is it convenient to check availability via the University Library System https://opac.uniba.it/easyweb/w8018/index.php? and contact the library to arrange the loan
Notes to the books	During the lectures, the teacher will illustrate the concepts with the help of slides that summarize (and sometimes integrate) the contents of the reference text. The slides will be made available at the end of each lesson on the online platform (see 'virtual venue' above). The following are available on the online platform: ● supporting slides used by the lecturer during lectures; ● lecturer's notes on the project required for the exam; ● Examples of solutions of the capture-the-flag (CTF) exercise on vulnerable machines at https://www.vulnhub.com/. This examples are a guide for students to the preparation of the project for the exam.

Organization of the didactic activities

Hours			
Total	Lectures	Project work	Individual study
hours	32 hours	50 hours	68 hours
CFU/ETCS			
CFU	4 CFU	2 CFU	

Teaching methods

	Classroom teaching with theory lectures.

Expected learning outcomes	
Knowledge and understanding	Students will gain the ability to understand the critical security issues involved in the design of software systems, taking on two perspectives: that of the malicious user, interested in attacking computer systems, and, speculatively, that of the knowledgeable developer, interested in mitigating existing security threats in systems during the development phase.
Applying knowledge and understanding	Students will be able to use the knowledge gained to identify vulnerabilities in a system and identify possible solutions (e.g., writing code more robust to prevent the possibility of attack)
Other skills	*Making judgements* Students will acquire the ability to assess the vulnerabilities of a software system and how to apply strategies that can prevent and mitigate possible attacks *Communication* Students will be able to describe the vulnerabilities of a software system and the decisions made to mitigate them. *Learning skills* Students will acquire the ability to learn the critical issues in the design of a software system from two points of view: the malicious user, interested in attacking the systems, and the developer, interested in mitigating the security threats existing in the systems during the development phase

Assessment	
Assessment methods	Written test + Project - Classroom written test, 9 questions (5 multiple-choice questions and 4 open questions) on theory regarding syllabus topics; allotted time 90 minutes; maximum score 33/33. The written test is considered passed if the student achieves a major equal score of 18/33. - Project: the student must send the project via institutional email at least one week from the date of the written text. The project consists of a report (in PDF or Word) that describes the solution of a Capture the Flag (CTF). The vulnerable machine will be chosen by the student and agreed with the teacher. In the project, the student will explain the choices made to solve the challenge (with screens of their pc), the vulnerabilities found (consistent with the content seen during the lectures), and the defensive techniques that could be adopted to prevent attacks.

	The CTF chosen for the project should be agreed upon in advance with the lecturer via institutional email. The teacher will evaluate the project and discuss it with the student during the written test. The project is valid only for the exam sessions held in the 2024-25 academic year. The project is considered passed if the student achieves a grade greater than 18/33 at the time of its discussion.
Evaluation criteria	Knowledge and understanding skills: o Ability to identify critical issues in a software system and identify solutions that can prevent and mitigate possible attacks. Applied knowledge and understanding skills: o Ability to describe, with real-world practical examples, the various attack and defense techniques adopted in software systems. Autonomy of judgment: Ability to decide which strategies to adopt when designing software systems. Communication skills: o Ability to describe, through the topics covered in the course, the choices made to strengthen the security of software systems during the development phase. Learning skills: o Understanding of the concepts learned during the course not only applied to the specific cases described during the course but also applicable to different contexts (e.g., software systems implemented in different programming languages or software systems based on Artificial Intelligence).
Measurements and final grade	The final grade is determined by the following formula 1/2*(written test grade) + 1/2*(project grade).
Further information	Students are advised to rely exclusively on the information/communications provided on the official websites of the Department of Computer Science, or on social groups only if established and administered exclusively by the teachers of the relevant courses: ● https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-di-laurea/corsi-di-laurea ● https://www.uniba.it/it/ricerca/dipartimenti/informatica ● https://elearning.di.uniba.it/ The teaching programs are available here: ● https://programmi.di.uniba.it/ The information that all students should know is written in the Teaching

Regulations and study posters available on the site:
● https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-di-laurea/corsi-di-laurea

Students are advised to be wary of information and materials circulating on unofficial sites or social groups, as they are often found to be unreliable, incorrect or incomplete. If you have any doubts, ask the teacher for a meeting according to the reception procedures.

_____

Students will be able to join the A.A. course forum. 2024/25 by enrolling in the course on the e-learning platform of the ADA department: https://elearning.uniba.it/