باسمه تعالی

طرح درس جهت ارائه در نیمسال دوم تحصیل ۱۴۰۳-۱۴۰۲

دانشکده	**مهندسی برق و کامپیوتر**	**گروه**	**معماری سیستم‌های کامپیوتری**
گرایش	**امنیت سایبری**	**مقطع**	**کارشناسی ارشد**
نام درس	**سیستم‌های نرم‌افزاری امن**	**نوع درس**	**پایه** ☐ **نظری** ☑ **تخصصی** ☑ **عملی** ☐ **اختیاری** ☑ **نظری-عملی** ☐
تعداد واحد	**۳**	**نام استاد**	**علیرضا شفیعی‌نژاد و حسین همائی**
دروس پیش‌نیاز	**ندارد**	**تلفن دفترکار**	**۵۰۹۴ و ۳۹۰۵**
دروس هم‌نیاز	**ندارد**	**پست الکترونیک**	shafieinejad@modares.ac.ir homaei@modares.ac.ir

✓ **اهداف درس:**

۱. آشنایی با چالش‌های طراحی و پیاده‌سازی سیستم‌های امن نرم‌افزاری
۲. آشنایی با نحوه تلفیق امنیت و چرخه حیات توسعه نرم‌افزار
۳. آشنایی با مدل‌سازی تهدیدات و مدیریت مخاطرات امنیتی
۴. آشنایی با آسیب‌پذیری‌های نرم‌افزاری و چگونگی بهره‌مندی مهاجمان از آن‌ها
۵. آشنایی با روش‌های آزمون‌های نرم‌افزار و آزمون‌های امنیتی
۶. آشنایی با مفاهیم کدنویسی امن
۷. آشنایی با انواع حملات سرقت کنترل و نفوذ به سیستم‌ها و مکانیزم‌های دفاعی در برابر آنها
۸. اصول معماری امن نرم‌افزار و سیاست دسترسی حداقلی
۹. آشنایی با جداسازی و مکانیزم Sandboxing
۱۰. امنیت برنامه‌های وب، آسیب‌پذیری‌ها و مکانیزم‌های دفاعی

✓ **رئوس مطالب و برنامه ارائه در کلاس:**

شماره هفته	موضوع جلسه درس	توضیحات
هفته اول	مقدمه	معرفی درس، اهمیت موضوع، اصول امنیت نرم‌افزار
هفته دوم	تلفیق امنیت و چرخه حیات توسعه نرم‌افزار	اصول کلی، آشنایی با بازبینی کد و تست نفوذ، اصول تحلیل و مدیریت مخاطرات امنیتی نرم‌افزار، تحلیل مقاومت در برابر حمله، تحلیل ابهام، تحلیل ضعف
هفته سوم	تلفیق امنیت و چرخه حیات توسعه نرم‌افزار	آزمون امنیتی مبتنی بر ریسک، نیازمندی‌های امنیتی، ضد نیازمندی، موارد سوء استفاده، استقرار امن
هفته چهارم	مدل‌سازی تهدید	اصول مدل‌سازی تهدید، STRIDE، درخت حمله، روش‌های مقابله با تهدیدات، اولویت‌بندی تهدیدات
هفته پنجم	طراحی امن نرم‌افزار اصول آزمون نرم‌افزار	روش‌های برآورده کردن محرمانگی، صحت، دسترس‌پذیری، تصدیق اصالت، مجازشناسی و ممیزی در طراحی نرم‌افزار اصول آزمون نرم‌افزار، انواع معیارهای آزمون، پوشش عبارات منطقی، ساختارهای نحوه و آزمون جهش، آزمون بر اساس مشخصه‌های دامنه ورودی

هفته ششم	آزمون نرم‌افزار	معیارهای پوشش ساختاری گراف، معیارهای پوشش جریان داده
هفته هفتم	آزمون ایستای کد	استخراج گراف از روی کد، گراف جریان کنترل و جریان داده، گراف فراخوانی، به‌کارگیری معیارهای پوشش گراف، پوشش‌های میان‌فرایندی
هفته هشتم	تولید ورودی آزمون	انواع فازینگ، اجرای نمادین، اجرای Concolic
هفته نهم	مفاهیم پایه کدنویسی امن	تاریخچه آسیب‌پذیری‌ها، معرفی فرصت‌ها و تهدیدها، بازارهای سفید و سیاه فروش آسیب‌پذیری‌های نرم‌افزار
هفته دهم	حملات سرقت کنترل	حملات سرریز بافر، سرریز اعداد صحیح، سرریز heap، ROP
هفته یازدهم	مکانیزم‌های دفاعی در برابر سرقت کنترل	مکانیزم‌های DEP، StackGaurd، ProPolice، SAFESEH/SEHSOP و LibSafe
هفته دوازدهم	جداسازی و Sandboxing	مفاهیم پایه مجازی‌سازی، Chroot & Jailkit، ptrace & systrace
هفته سیزدهم	جداسازی و Sandboxing	System call Interposition، جداسازی مبتنی بر ماشین مجازی، Software Fault Isolation
هفته چهاردهم	اصول معماری امن	اصول حداقل دسترسی، کنترل دسترسی و به کارگیری کنترل‌های امنیتی سیستم‌عامل
هفته پانزدهم	امنیت برنامه‌های وب ۱	امنیت مرورگر، امنیت وب سرور، سیاست امنیتی محتوامحور CSP، حملات تزریق کد، XSS، CSRF و مکانیزم‌های مقابله با آنها
هفته شانزدهم	امنیت برنامه‌های وب ۲	مدیریت نشست و احراز اصالت کاربران، به کارگیری صحیح HTTPS

✓ **روش ارزشیابی:**

سمینار (۲ نمره)، تمرین‌ها (۲ نمره)، پروژه‌ها (۲ نمره)، امتحانات (۱۴ نمره)

✓ **منابع :**

[1] G. McGraw, *Software Security: Building Security In*, Addison-Wesley, 2006.
[2] Paul, Mano. *Official (ISC)^2 guide to the CSSLP CBK*. 2nd edition, CRC Press, 2014.
[3] A. Shostack, *Threat Modeling: Designing for Security*, Wiley, 2014.
[4] Ammann, Paul, and Jeff Offutt. *Introduction to software testing*, 2nd ed. Cambridge University Press, 2016.
[5] A. Zeller, R. Gopinath, M. Böhme, G. Fraser, and C. Holler, *The Fuzzing Book*, 2024. [Online]. Available: https://www.fuzzingbook.org/.
[6] S. Parsa, *Software Testing Automation, Testability Evaluation, Refactoring, Test Data Generation and Fault Localization*, Springer, 2024.
[7] A. Sotirov, Heap Feng Shui in Javascript, *Blackhat Europe*, 2007.
[8] M. Daniel, J. Honoroff, and C. Miller, Engineering Heap Overflow Exploits with JavaScript, Proceedings of the 2nd conference on USENIX Workshop on offensive technologies (WooT), 2008.
[9] P. Ratanaworabhan, B. Livshits, and B. Zorn**,** Nozzle: A Defense Against Heap-spraying Code Injection Attacks, USENIX security symposium. 2009.
[10] Dion Blazakis, Interpreter Exploitation: Pointer inference and JiT spraying. BlackHat, 2010.