ĐỀ TÀI ĐỒ ÁN MÔN HỌC
MÔN HỌC: LẬP TRÌNH AN TOÀN VÀ KHAI THÁC LỖ HỔNG PHẦN MỀM
Học Kỳ I - Năm học 2023 - 2024

Lớp: NT521
Yêu cầu:
Lớp Tài năng: **tối đa 3 SV/nhóm**
Lớp CLC: tối đa **4 SV/nhóm**
Lớp ANTT: tối đa **5 SV/nhóm**

**** Lưu ý:
+ Sinh viên xem trước đề tài đồ án CK và lựa chọn đề tài mong muốn để đăng ký trên moodle.
+ Đọc trước Phần tóm tắt nội dung của tài liệu để biết trước nội dung chính của đề tài.
+ Hình thức giao đề tài: SV đăng ký trên moodle, mỗi đề tài được thực hiện bởi tối đa 1 nhóm SV.
+ Cách thực hiện: Tìm hiểu lý thuyết về ngữ cảnh, mô hình/phương pháp/cách thực nghiệm được sử dụng trong tài liệu của đề tài; và thực hiện demo về một trong các nội dung được đề cập trong đề tài
+ Vắng buổi báo cáo đồ án môn học của nhóm sẽ bị chấm điểm 0 cho điểm Đồ án.
+ Mỗi nhóm trình bày trong 15 phút (thuyết trình +demo) - không cho phép trình bày vượt quá thời gian. Sau đó, hỏi-đáp trong vòng 5 phút.

Mã đề tài	**Tên đề tài**	**Link tham khảo triển khai (bài báo khoa học)** *(SV dùng tên đề tài để tìm tài liệu tương ứng/và code mẫu github (nếu có))*	**Ghi chú**
CK01	Probabilistic Path Prioritization for Hybrid Fuzzing	https://ieeexplore.ieee.org/document/9280412	
CK02	Software Crash Analysis for Automatic Exploit Generation on Binary Programs	https://ieeexplore.ieee.org/abstract/document/6717039	
CK03	jTrans: jump-aware transformer for binary code similarity detection	https://dl.acm.org/doi/abs/10.1145/3533767.3534367	
CK04	HAEPG: An Automatic Multi-hop Exploitation Generation Framework	https://link.springer.com/chapter/10.1007/978-3-030-52683-2_5	
CK05	Revery: From Proof-of-Concept to Exploitable	https://wcventure.github.io/FuzzingPaper/Paper/CCS18_Revery.pdf	
CK06	Vulnerability-oriented directed fuzzing for binary programs	https://www.nature.com/articles/s41598-022-07355-5	
CK07	BofAEG: Automated Stack Buffer Overflow Vulnerability Detection and Exploit Generation Based on Symbolic Execution and Dynamic Analysis	https://dl.acm.org/doi/abs/10.1155/2022/1251987	

Mã đề tài	Tên đề tài	**Link tham khảo triển khai (bài báo khoa học)** *(SV dùng tên đề tài để tìm tài liệu tương ứng/và code mẫu github (nếu có))*	Ghi chú
CK08	ProFuzzer: On-the-fly Input Type Probing for Better Zero-Day Vulnerability Discovery	https://youwei1988.github.io/papers/SP2019.pdf	
CK09	VulDetector: Detecting Vulnerabilities Using Weighted Feature Graph Comparison	https://ieeexplore.ieee.org/document/9309254	
CK10	BinDeep: A deep learning approach to binary code similarity detection	https://www.sciencedirect.com/science/article/pii/S0957417420310332	
CK11	CD-VulD: Cross-Domain Vulnerability Discovery Based on Deep Domain Adaptation	https://ieeexplore.ieee.org/document/9054952	
CK12	µµVulDeePecker: A Deep Learning-Based System for Multiclass Vulnerability Detection	Paper: https://ieeexplore.ieee.org/abstract/document/8846081 Github NDSS 2018: https://github.com/CGCL-codes/VulDeePecker An extension of VulDeePecker: “SySeVR: A Framework for Using Deep Learning to Detect Software Vulnerabilities” (https://arxiv.org/abs/1807.06756)	
CK13	Arbiter: Bridging the Static and Dynamic Divide in Vulnerability Discovery on Binary Programs	https://www.usenix.org/conference/usenixsecurity22/presentation/vadayath Github: https://github.com/jkrshnmenon/arbiter	
CK14	Binary-level Directed Fuzzing for Use-After-Free Vulnerabilities	https://www.usenix.org/system/files/raid20-nguyen.pdf	
CK15	Large-Scale Empirical Study of Important Features Indicative of Discovered Vulnerabilities to Assess Application Security	https://ieeexplore.ieee.org/document/8629314	
CK16	SySeVR: A Framework for Using Deep Learning to Detect Software Vulnerabilities	Github: https://github.com/SySeVR/SySeVR	
CK17	HyVulDect: A hybrid semantic vulnerability mining system based on graph neural network	https://www.sciencedirect.com/science/article/abs/pii/S0167404822002176	
CK18	Deep Learning based Vulnerability Detection: Are We There Yet?	Github: https://github.com/VulDetProject/ReVeal Paper: https://par.nsf.gov/servlets/purl/10326688	
CK19	KOOBE: Towards Facilitating Exploit Generation of Kernel Out-Of-Bounds Write Vulnerabilities	https://www.usenix.org/conference/usenixsecurity20/presentation/chen-weiteng	

Mã đề tài	Tên đề tài	**Link tham khảo triển khai (bài báo khoa học)** *(SV dùng tên đề tài để tìm tài liệu tương ứng/và code mẫu github (nếu có))*	Ghi chú
CK20	FUZE: Towards Facilitating Exploit Generation for Kernel Use-After-Free Vulnerabilities	https://www.usenix.org/conference/usenixsecurity18/presentation/wu-wei	
CK21	Asteria-Pro: Enhancing Deep-Learning Based Binary Code Similarity Detection by Incorporating Domain Knowledge	https://dl.acm.org/doi/abs/10.1145/3604611	
CK22	Function Representations for Binary Similarity	https://ieeexplore.ieee.org/document/9325042	
CK23	Multi-semantic feature fusion attention network for binary code similarity detection	https://www.nature.com/articles/s41598-023-31280-w	
CK24	LineVD: statement-level vulnerability detection using graph neural networks	https://dl.acm.org/doi/abs/10.1145/3524842.3527949	
CK25	CSGVD: A deep learning approach combining sequence and graph embedding for source code vulnerability detection	https://www.sciencedirect.com/science/article/abs/pii/S0164121223000183	
CK26	VulChecker: Graph-based Vulnerability Localization in Source Code	https://www.usenix.org/system/files/sec23summer_449-mirsky-prepub.pdf	
CK27	CPVD: Cross Project Vulnerability Detection Based on Graph Attention Network and Domain Adaptation	https://ieeexplore.ieee.org/document/10149539	
CK28	FLAG: Finding Line Anomalies (in code) with Generative AI	https://arxiv.org/pdf/2306.12643.pdf	
CK29	Compact Abstract Graphs for Detecting Code Vulnerability with GNN Models	https://dl.acm.org/doi/abs/10.1145/3564625.3564655	
CK30	LineVul: A Transformer-based Line-Level Vulnerability Prediction	https://dl.acm.org/doi/10.1145/3524842.3528452	
CK31	VDoTR: Vulnerability detection based on tensor representation of comprehensive code graphs	https://www.sciencedirect.com/science/article/abs/pii/S0167404823001578	
CK32	Detecting Condition-Related Bugs with Control Flow Graph Neural Network	https://dl.acm.org/doi/abs/10.1145/3597926.3598142	
CK33	Combining Graph-Based Learning With Automated Data Collection for Code Vulnerability Detection	https://ieeexplore.ieee.org/document/9293321; Github: https://github.com/HuantWang/FUNDED_NISL	