


THÔNG TIN CHUNG CỦA BÁO CÁO

- Link YouTube video của báo cáo (tối đa 5 phút): <https://youtu.be/kbm-rwZyT5A>
- Link slides (dạng .pdf đặt trên Github):
<https://github.com/cometofruition/CS2205.CH1702/blob/main/Trung%20%C4%90o%C3%A0n%20Minh%20-%20Slide.pdf>
- *Mỗi thành viên của nhóm điền thông tin vào một dòng theo mẫu bên dưới*
- *Sau đó điền vào Đề cương nghiên cứu (tối đa 5 trang), rồi chọn Turn in*

<ul style="list-style-type: none">• Họ và Tên: Đoàn Minh Trung• MSHV: 220202016 	<ul style="list-style-type: none">• Lớp: CS2205.CH1702 - APR2023• Tự đánh giá (điểm tổng kết môn): 8.5/10• Số buổi vắng: 1• Số câu hỏi QT cá nhân: 0• Số câu hỏi QT của cả nhóm: 0• Link Github: https://github.com/mynameuit/CS2205.APR2023/• Mô tả công việc:<ul style="list-style-type: none">○ Lên ý tưởng cho đề tài○ Viết đề cương○ Làm video YouTube
---	--

ĐỀ CƯƠNG NGHIÊN CỨU

TÊN ĐỀ TÀI: TÌM HIỂU PHƯƠNG PHÁP PHÁT SINH MÃ ĐỘC ANDROID BẰNG MẠNG SINH ĐỐI KHÁNG
TÊN ĐỀ TÀI TIẾNG ANH: A STUDY ON ANDROID MALWARE GENERATION BY USING GENERATIVE ADVERSARIAL NETWORKS

TÓM TẮT (Tối đa 400 từ)

Nghiên cứu và phát triển các phương pháp phát sinh mã độc đối kháng, tập trung vào việc tìm hiểu và dựa trên những hạn chế của các công trình nghiên cứu trước đây để đưa ra giải pháp mới trong cách phát sinh AE bằng cách ánh xạ đến không gian đặc trưng có thể thay đổi của các ứng dụng lành tính nhằm tăng cường độ tương đồng giữa AE và ứng dụng lành tính, tuy nhiên vẫn giữ lại các đặc trưng ban đầu để duy trì các chức năng, khả năng thực thi và hành vi độc hại của ứng dụng mã độc gốc. Mở rộng bộ dữ liệu huấn luyện và thử nghiệm bằng cách bổ sung các mẫu mã độc và AE có tính chất đa dạng và phong phú.

GIỚI THIỆU (Tối đa 1 trang A4)

Với sự phát triển không ngừng và nhanh chóng của mã độc trên thiết bị di động, đặc biệt là trên nền tảng Android, việc xây dựng hệ thống phát hiện mã độc dựa trên học máy (Machine Learning - ML) đã trở thành một ưu tiên quan trọng và thu hút sự quan tâm lớn từ cộng đồng nghiên cứu trên toàn cầu. Mặc dù các thuật toán học máy hiện nay đã có những bước tiến quan trọng trong việc phát hiện mã độc, nhưng chúng vẫn dễ bị ảnh hưởng bởi các cuộc tấn công đối kháng (Adversarial example - AE) [1]-[3]. Chỉ cần thêm vào một số lượng nhiễu (noise) nhỏ, cụ thể là sửa đổi một số ít các đặc trưng khác nhau từ tập tin kê khai (manifest file) của ứng dụng, các cuộc tấn công AE có thể dễ dàng khiến hệ thống phát hiện mã độc Android phân loại nhầm thành các ứng dụng lành tính (benign). Tuy nhiên, có một vấn đề mà phần lớn các nghiên cứu đang mắc phải đó là họ chỉ tập trung vào việc cải thiện khả năng đánh lừa và hiệu suất phát hiện các cuộc tấn công đối kháng, mà không chú trọng đến việc duy trì chức năng, khả năng thực thi và hành vi ban đầu của chúng [4] [5]. Do tính phức tạp về mặt lập trình nên việc xáo trộn (perturbation), thêm nhiễu hay sửa đổi các đặc trưng ngữ nghĩa (semantic feature) được thu thập từ Dalvik bytecode (tức là tập tin class.dex) mà không ảnh hưởng đến chức năng ban đầu là vô cùng khó khăn, ngay cả những sửa đổi nhỏ. Một nghiên cứu trước đây đã chứng minh rằng việc xáo trộn các đặc trưng cú pháp (syntactic feature) như quyền truy cập (permission) của ứng dụng Android dễ dàng duy trì được chức năng ban đầu hơn. Có một nghiên cứu khác cũng đề xuất một số thuật toán và công cụ tự động để tính toán nhiễu, tập trung sửa đổi các đặc trưng (cú pháp hoặc ngữ nghĩa) không gây ảnh hưởng đến chức năng của ứng dụng mã độc gốc [6] [7]. Mặc dù các nghiên cứu này cũng như nhiều nghiên cứu khác đã khám phá ra nhiều giải pháp để tạo AE mà vẫn

duy trì được hành vi độc hại ban đầu, nhưng họ vẫn chưa đưa ra bất kỳ kết quả thử nghiệm nào để chứng minh rằng các AE này sau khi được đóng gói lại (repackage), sẽ hoạt động giống như ứng dụng mã độc gốc. Vì vậy, trong định hướng của tôi, tôi sẽ tập trung nghiên cứu trên bài toán này bắt đầu từ việc xây dựng hệ thống phát sinh mã độc Android bằng Mạng sinh đối kháng (Generative Adversarial Networks – GAN). Các cuộc tấn công này phát sinh AE dựa trên phương pháp tính toán độ dốc (gradient) của hàm mất mát (loss function) cho dữ liệu đầu vào và dựa trên độ biến dạng (distortion) nhỏ nhất.

Ngoài ra, tôi cũng tập trung sửa đổi các phương pháp tạo AE bằng cách vô hiệu hóa các xáo trộn trên các đặc trưng không thể thay đổi và chỉ sửa đổi các đặc trưng không gây ảnh hưởng đến hành vi độc hại của ứng dụng mã độc. Mục đích là để tăng tỉ lệ và số lượng ứng dụng mã độc đối kháng duy trì chức năng ban đầu thành công. Từ đó, nghiên cứu, mở rộng và nâng cao chất lượng một bộ dữ liệu huấn luyện mới nhằm cải thiện khả năng nhận diện cho các hệ thống phát hiện mã độc Android.

Phương pháp phát sinh mã độc đối kháng bằng học máy là bài toán đòi hỏi sử dụng các thuật toán và phương pháp tính toán độ dốc của hàm mất mát để thay đổi các đặc trưng của ứng dụng mã độc một cách tối ưu, nhằm tránh ảnh hưởng đến chức năng, khả năng thực thi cũng như hành vi độc hại ban đầu. Trong đề tài này, bài toán của chúng tôi được mô tả như sau:

Cho một ứng dụng mã độc X_m mà các hệ thống đã có khả năng phát hiện chúng. Mục tiêu là xây dựng một hệ thống có thể phát sinh mã độc đối kháng X_{adv} bằng GAN (G), cụ thể:

o **Đầu vào (Input):** Ứng dụng mã độc gốc X_m có nhãn phân loại $Y \in \{0, 1\}$ (trong đó 0 biểu thị ứng dụng lành tính X_b và 1 biểu thị ứng dụng mã độc X_m) và thông số phương pháp phát sinh mã độc G.

o **Đầu ra (Output):** Mã độc đối kháng được tạo ra X_{adv} , trong đó $G(X_m) = X_{adv}$ với nhãn phân loại $Y = 0$.

MỤC TIÊU

- Tiến hành xây dựng một hệ thống phát sinh mã độc đối kháng, có khả năng gây nhầm lẫn và khiến cho các mô hình phát hiện mã độc phân loại các mẫu mã độc thành lành tính.

- Các mẫu có tỉ lệ đánh lừa thành công cao sẽ được hệ thống này tự động đóng gói lại thành các ứng dụng mã độc mới và được đem đi kiểm định xem có còn duy trì được các chức năng, khả năng thực thi và hành vi ban đầu không.
- Tăng số lượng các ứng dụng mã độc đối kháng duy trì được các chức năng, khả năng thực thi và hành vi ban đầu, tiến hành xây dựng mô hình phát sinh mới dựa trên các đặc trưng có thể thay đổi được và giữ lại các đặc trưng không thể thay đổi để bảo toàn các hành vi độc hại của chúng.

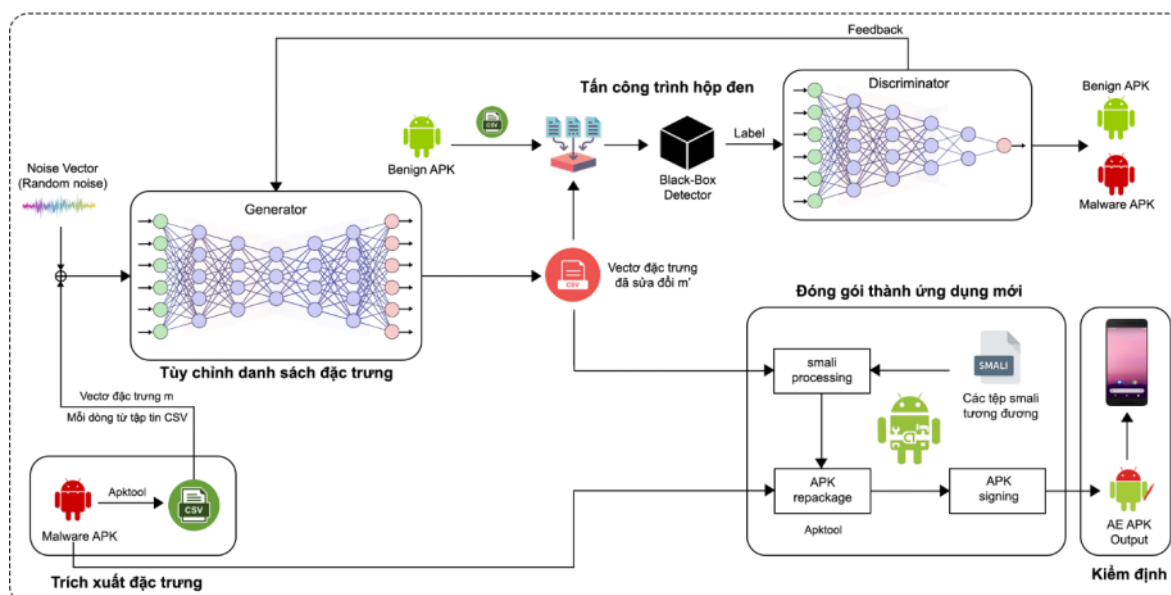
NỘI DUNG VÀ PHƯƠNG PHÁP

(Viết nội dung và phương pháp thực hiện để đạt được các mục tiêu đã nêu)

Nội dung 1: Xây dựng hệ thống phát sinh mã độc đối kháng dựa trên học máy

Phương pháp

Mục tiêu của tôi là sử dụng GAN sinh ra bộ dữ liệu gồm 21,000 mẫu ứng dụng mã độc đối kháng được thu thập từ bộ dữ liệu AndroZoo có tỉ lệ vượt mặt thành công các mô hình phát hiện mã độc trên 90%. Sau đó, kiểm định và đánh giá khả năng bảo toàn chức năng và hành vi độc hại của bộ dữ liệu này. Hệ thống này bao gồm 5 bước chính và quy trình được thể hiện rõ ở Hình 2 như sau:



Hình 2 – Quy trình xây dựng hệ thống phát sinh mã độc bằng học máy.

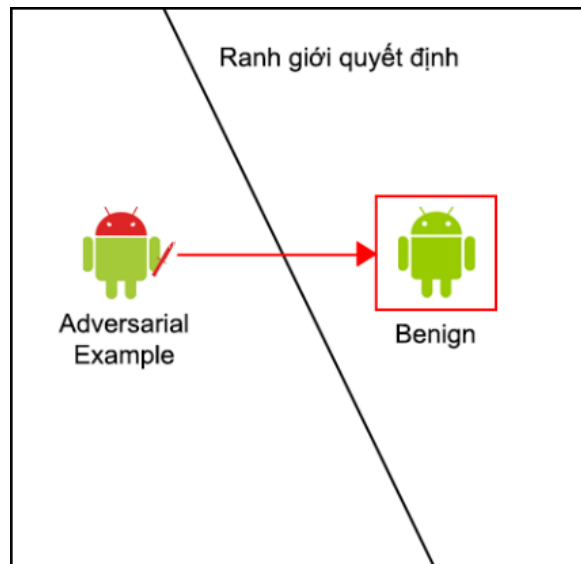
Nội dung 2: Nghiên cứu phương pháp phát sinh mã độc đối kháng mới dựa trên các

đặc trưng có thể thay đổi

Phương pháp

Thực hiện sửa đổi các phương pháp phát sinh AE bằng cách sau:

- Vô hiệu hóa các xáo trộn trên các đặc trưng không thể thay đổi sau đó ánh xạ đến tập đặc trưng có thể phân biệt được. Trong một ứng dụng mã độc Android, những đặc trưng không thể thay đổi ví dụ như permissions, quyền truy cập vào camera của thiết bị, đọc danh bạ, gửi SMS,... Sau khi ứng dụng đã khai báo các quyền này, chính ứng dụng sẽ không thể sửa đổi hoặc loại bỏ chúng đi. Người dùng phải cấp các quyền này trong quá trình cài đặt và việc loại bỏ có thể ảnh hưởng đến chức năng của ứng dụng.
- Sửa đổi một số lượng nhỏ các đặc trưng chức năng có thể thay đổi trên, giới hạn khoảng cách giữa X_{adv} và X_m nhỏ nhất và làm cho X_{adv} nằm gần ranh giới quyết định (decision boundary) với tập X_b (Hình 3).



Hình 3 - Mô phỏng xu hướng mẫu mã độc đối kháng nằm gần ranh giới quyết định với mẫu lành tính.

Nội dung 3: Tăng cường bộ dữ liệu huấn luyện với các mẫu mã độc đối kháng.

Phương pháp

Sau khi phát sinh ra các đặc trưng đối kháng theo phương pháp cải thiện sự bảo toàn chức năng của tôi ở nội dung 2, tôi sẽ tiến hành lặp lại các bước 3-5 ở nội dung một để tiếp tục thử nghiệm tấn công vào các mô hình phát hiện mã độc dựa trên học máy. Sau đó đóng gói thành các ứng dụng mã độc và đem đi kiểm định. Từ đó, cho ra kết quả so sánh

tỉ lệ thành công so với các phương pháp trước.

KẾT QUẢ MONG ĐỢI

(Viết kết quả phù hợp với mục tiêu đặt ra, trên cơ sở nội dung nghiên cứu ở trên)

- Hệ thống phát sinh mã độc đối kháng dựa trên học máy đạt tỉ lệ tấn công thành công cao vào các mô hình phát hiện.
- Dự kiến tỉ lệ thành công các mẫu mã độc đối kháng duy trì được chức năng vào khoảng >50%.

TÀI LIỆU THAM KHẢO *(Định dạng DBLP)*

- [1]. Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy, ‘Explaining and harnessing adversarial examples’, in International Conference on Learning Representations, (2015).
- [2]. Juncheng Li, Frank R. Schmidt, and J. Zico Kolter, ‘Adversarial camera stickers: A physical camera-based attack on deep learning systems’, in The International Conference on Machine Learning, pp. 3896–3904, (2019).
- [3]. Xingxing Wei, Siyuan Liang, Ning Chen, and Xiaochun Cao, ‘Transferable adversarial attacks for image and video object detection’, in IJCAI, pp. 954–960, (2019).
- [4]. Zhangjie Fu, Yongjie Ding, and Musaazi Godfrey. An lstm-based malware detection using transfer learning. Journal of Cybersecurity,3(1):11, 2021.
- [5]. Jianhua Wang, Xiaolin Chang, Yixiang Wang, Ricardo J Rodríguez, and Jianan Zhang. Lsgan-at: enhancing malware detector robustness against adversarial examples. Cybersecurity, 4:1–15, 2021.
- [6]. Xiao Chen, Chaoran Li, Derui Wang, Sheng Wen, Jun Zhang, Surya Nepal, Yang Xiang, and Kui Ren. Android hiv: A study of repackaging malware for evading machine-learning detection. IEEE Transactions on Information Forensics and Security, 15:987–1001, 2020.
- [7]. Hamid Bostani and Veelasha Moonsamy. Evadedroid: A practical evasion attack on machine learning for black-box android malware detection. arXiv preprint arXiv:2110.03301, 2021.