**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**NHÓM 4**

**BÁO CÁO ĐỒ ÁN GIỮA KÌ**

**Forensic analysis and detection of Android Ransomware**

**MÔN : PHÁP CHỨNG KĨ THUẬT SỐ**

**LỚP : NT334.O11.ANTT**

**MỤC LỤC**

[Chương 1. Thực nghiệm 2](#_Toc154404663)

[Chương 2. Demo 4](#_Toc154404664)

[2.1. Phân tích tĩnh với 4](#_Toc154404665)

[2.2. Phân tích động với DroidBox 4](#_Toc154404666)

[2.3. Áp dụng machine learning 8](#_Toc154404667)

[2.3.1. Dataset 8](#_Toc154404668)

[2.3.2. Kết quả Áp dụng Machine learning 9](#_Toc154404669)

# Thực nghiệm

Các phương pháp forensic hiện tại:

Dynamic

Mặt khác, phân tích động, kiểm tra ứng dụng trong quá trình thực hiện. Một số phần mã có thể bỏ lỡ bằng phân tích tĩnh có thể được phát hiện trong phân tích động. Phân tích động dựa trên quan sát thời gian chạy về việc sử dụng bộ nhớ, sử dụng CPU, hành vi mạng, cũng như kiểm tra các cuộc gọi hệ thống của ứng dụng. Ưu điểm chính của kỹ thuật này là nó phát hiện tải mã động và ghi lại hành vi ứng dụng trong thời gian chạy, điều này bị cản trở trong phân tích tĩnh.

Một số nghiên cứu liên quan trong việc sử dụng hướng phân tích động trong việc phát hiện ransomeware nói riêng và malware nói chung:

**Uncovering the Face of Android Ransomware: Characterization and Real-Time Detection**

Trong bài viết này, tác giả tập trung vào nền tảng Android và nhằm mục đích mô tả ransomware Android hiện có. Cụ thể, tác giả đã quản lý để thu thập 2.721 mẫu ransomware bao gồm phần lớn các họ ransomware Android hiện có. Dựa trên các mẫu này, tác giả mô tả chúng một cách có hệ thống từ một số khía cạnh, bao gồm dòng thời gian và các tính năng độc hại. Ngoài ra, kết quả phát hiện của các công cụ chống vi-rút hiện tại khá đáng thất vọng, điều này rõ ràng đòi hỏi các giải pháp chống ransomware di động tùy chỉnh. Để phát hiện phần mềm tống tiền tống tiền người dùng bằng cách mã hóa dữ liệu, tác giả đề xuất một hệ thống phát hiện thời gian thực mới, được gọi là RansomProber. Bằng cách phân tích các widget giao diện người dùng của các hoạt động liên quan và tọa độ chuyển động ngón tay của người dùng, RansomProber có thể suy ra liệu các hoạt động mã hóa tệp có được bắt đầu bởi người dùng hay không. Kết quả thử nghiệm cho thấy RansomProber có thể phát hiện hiệu quả mã hóa ransomware với độ chính xác cao và hiệu suất thời gian chạy chấp nhận được.

**A screenshot of a phone

Description automatically generated**

Giao diện mẫu của simplocker

**Android Ransomware Detection Based on Dynamic Obtained Features**

Bài báo đã đề xuất phương pháp phát hiện ransomware Android bằng kỹ thuật phân tích động. Cách tiếp cận được đề xuất sử dụng các system call làm các tính năng thu được từ phân tích động thông qua công cụ strace của Linux. Các thuật toán phân loại Random Forest, J48 và Naïve Bayes đã được sử dụng để phân loại các trường hợp dựa trên các tính năng được đề xuất. Kết quả thí nghiệm cho thấy Randomforest đạt độ chính xác phát hiện cao nhất là 98,31% với tỷ lệ dương tính giả thấp nhất là 0,016

**A screenshot of a video game

Description automatically generated**

Quá trình thực thi file APK để trích xuất các lời gọi hệ thống.

# Demo

## Phân tích tĩnh với

## Phân tích động với DroidBox

Để cài đặt và sử dụng Droidbox, hệ điều hành yêu cầu để làm máy host là Linux hoặc MacOS, đồng thời đã cài đặt bộ ứng dụng Android SDK để sử dụng tính năng tạo máy ảo Android Virtual Device (AVD).Yêu cầu cấu hình máy AVD mới bao gồm : hệ điều hành: Android 4.1.2 , thiết bị: Nexus 4, Kiến trúc CPU: ARM.

Để thực thi được Droibox, trước tiên ta cần khởi chạy trình emulator của Android SDK với máy AVD đã được khởi tạo. Sau đó thực thi lệnh sau tại thư mục Droidbox:

./droidbox.sh <file.apk> <duration in secs (optional)>

Với file.apk là đường dẫn đến file APK, ngoài ra ta có thể set thời gian phân tích tối đa cho Droidbox bằng thêm duration, được tính bằng s.

Screens screenshot of a computer screen

Description automatically generated

Đối với việc tự động việc phân tích nhiều file APK, DroidBox không có hỗ trợ cho tác vụ trên. Do đó, nhóm tạo ra một script Python nhằm có thể tự động load emulator và phân tích các file APK lần lượt trong thư mục. Chi tiết script Python như sau :

A screenshot of a computer program

Description automatically generated

Đầu tiên script sẽ tìm kiếm các file apk có trong đường dẫn thư mục được chỉ định và lưu lại tất cả các đường dẫn đến mỗi file.

Sau đó, lần lượt duyệt qua các đường dẫn của file để thực hiện việc phân tích file đó bằng Droibox.

Quá trình chạy phân tích như sau:

* Đầu tiên, thực thi file emulator mặc định của Android SDK với -avd là tên máy ảo đã khởi tạo trước đó, đồng thời thêm các option -wipe-data để xóa toàn bộ dữ liệu, -prop Dalvik.vm.execution-mode=int:portable& nhằm cho phép DroidBox có thể truy cập được máy ảo đó.
* Sau đó, tạm dừng thực thi trong khoản 120s nhằm đợi cho trình emulator có thể load được máy avd thành công.Khoản thời gian này có thể tùy chỉnh tùy vào tốc độ của trình emulator.
* Khi emulator đã load được máy avd, thực thi lệnh gọi droidbox.sh với đường dẫn file apk đã load được ở trên, đồng thời set duration với khoản thời gian mong muốn để Droidbox có thể kết thúc việc phân tích và trả về kết quả. Tiếp tục tạm dừng thực thi trong khoản thời gian Droidbox phân tích.
* Sau khi DroidBox đã trả về kết quả và quá trình thực thi tiếp tục, tìm tiến trình có tên “qemu” và tắt tiến trình đó đi, việc này sẽ tắt đi trình emulator hiện tại. Tạm dừng thực thi để đợi emulator tắt thành công.
* Quay lại bước đầu tiên với file APK tiếp theo.

Với đoạn script trên, việc thực thi tự động phân tích các file APK trong 1 thư mục có thể thực hiện được.

Để có thể trích xuất được kết quả thực thi của Droidbox ra file csv, ta có thể thay đổi nội dung ở cuối file droidbox.py tại thư mục script của thư mục Droidbox như sau :

A screenshot of a computer

Description automatically generated

Một số kết quả phân tích mẫu như sau ;

A screen shot of a computer

Description automatically generated

## Áp dụng machine learning

Do dữ liệu từ 2 nguồn trên không đủ để thực hiện việc áp dụng machine learning trong việc phát hiện ransomware, nhóm tìm kiếm từ nguồn dataset khác có bộ dữ liệu có các feature cả static và dynamic.

### Dataset

Bộ dữ liệu được sử dụng thuộc về Tập dữ liệu KronoDroid, tập dữ liệu KronoDroid là một tập dữ liệu lai dựa trên thời gian được sử dụng để phát hiện và mô tả phần mềm độc hại Android hiệu quả. Nó bao gồm hai bộ dữ liệu phụ có tính năng như nhau: một từ các thiết bị thực và một từ trình giả lập. Tạo một tập dữ liệu bao gồm khung thời gian lớn hơn và 489 feature từ phân tích tĩnh và động được thu thập

Dataset đươc sử dụng để thử nghiệm được lấy từ phần real\_device, là kết quả phân tích tĩnh và phân tích động của :

* + 41,382 malware samples (240 malware families)
  + 36,755 benign apps.

Link dataset : [*https://github.com/aleguma/kronodroid*](https://github.com/aleguma/kronodroid)

Link bài báo của Dataset [*https://www.sciencedirect.com/science/article/pii/S0167404821002236*](https://www.sciencedirect.com/science/article/pii/S0167404821002236)

Các feature có trong dataset:

A screenshot of a document

Description automatically generated

### Kết quả Áp dụng Machine learning

Model Unsupervised learning sử dụng Variance Threshold technique và Gaussian Mixture Model

A screenshot of a computer code

Description automatically generated

A yellow and blue squares with black text

Description automatically generated

Supervised learning sử dụng Variance Threshold technique và Support Vector Machine

A screenshot of a computer code

Description automatically generated

A chart with different colored squares

Description automatically generated