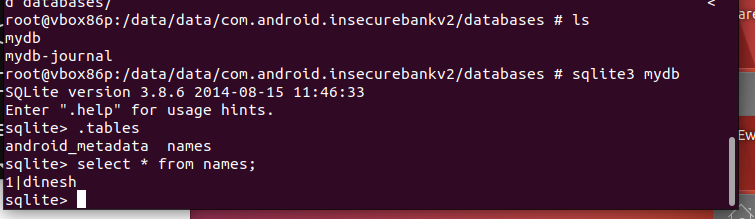
Yêu cầu 1 Phân tích và chỉ ra điểm bất thường của đoạn code trên?

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, số

Mô tả được tạo tự động

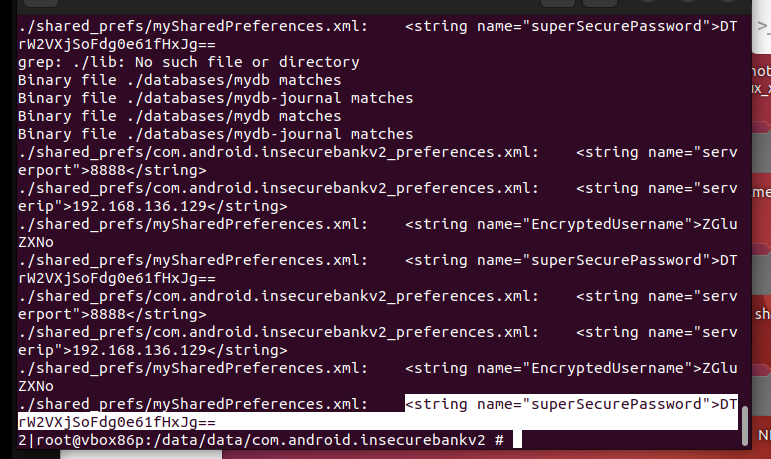
* Đây là đoạn code trong phần mềm insecurebank khi sử dụng bytecode viewer để xem
* Ở đây ta có thể thấy đoạn mã sử dụng phương thức HttpPost để có thể POST request đăng nhập của ta
* Do phương thức Http không có mã hóa nên thông tin đăng nhập của ta được lưu dưới dạng plaint text nên có thể dễ dạng nhìn thấy nếu có thể bắt được gói tin

Yêu cầu 2 Chỉ ra rằng dữ liệu lưu trữ có an toàn hay không?



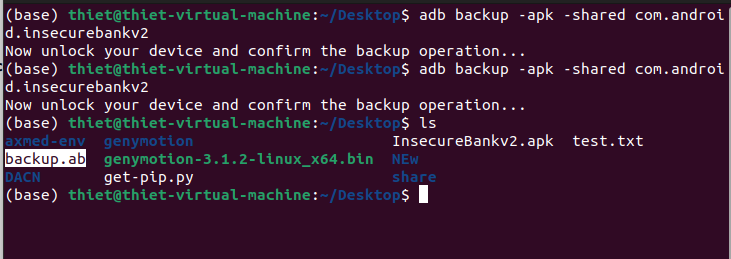
* Ta sử dụng adb để vào shell của điện thoại
* Vào đường dẫn như trên
* Ta thấy có file là my-db của ứng dụng insecurebankv2
* Truy cập vào và xem các tables thì ta thấy có tables là names
* select hết các tên trong trong tables thì ta thấy tên người dùng để đăng nhập nhưng nó không được mã hóa được lưu dưới dạng plaint text

Yêu cầu 3 Kiểm tra xem thông tin nhạy cảm có lưu lại trên thiết bị hay không? Một số từ khoá: deviceId, userId, imei, deviceSerialNumber, devicePrint, phone, XDSN, mdn, IMSI, uuid…

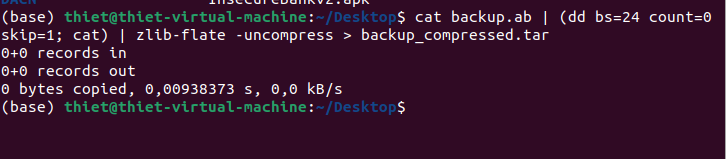


* khi sử dụng command “grep -r "name" $(find)”
* Thì ta tìm thấy được có 1 chuỗi user tên là “superSecurePassword” và chuỗi này có vẻ đã được mã hóa
* Và cũng có các thông tin nhạy cảm khác như là serverPort, serverIP, …

Yêu cầu 4 Theo bạn thư mục sao lưu chứa thông tin nào cần mã hoá, chỉ ra



* Sau khi backup thì ta có được file backup.ab



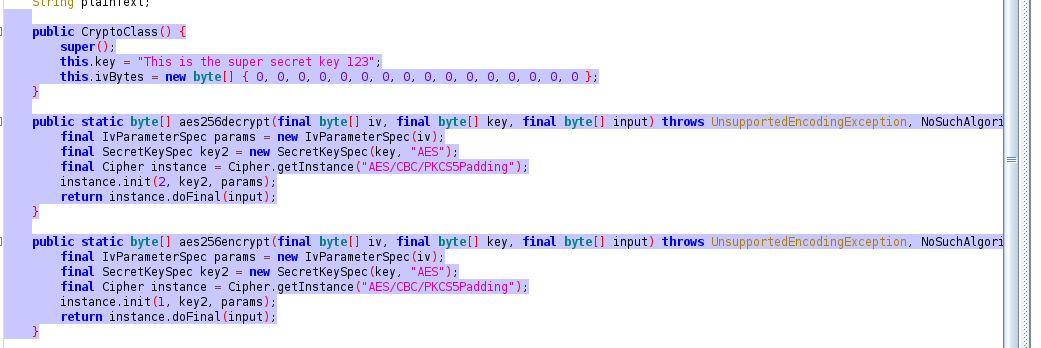
* Chuyển đổi tập tin sao lưu qua định dạng có thể đọc được bằng command ” cat backup.ab | (dd bs=24 count=0 skip=1; cat) | zlib-flate -uncompress > backup\_compressed.tar”

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Trang web

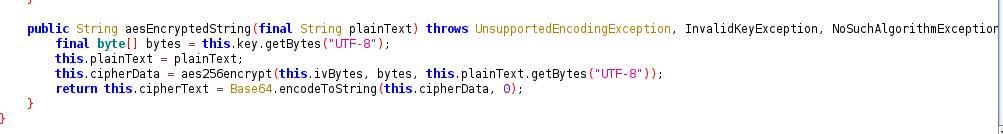
Mô tả được tạo tự động

* Vì người dùng đã đăng nhập vào app trước khi sao lưu nên đó đã được lưu tạo file mysharePreferences.xml
* Nhưng những thông tin này đã được mã hóa nên em nghĩ là những thông tin cần sao lưu lại là username, password đều cần được mã hóa trước khi sao lưu

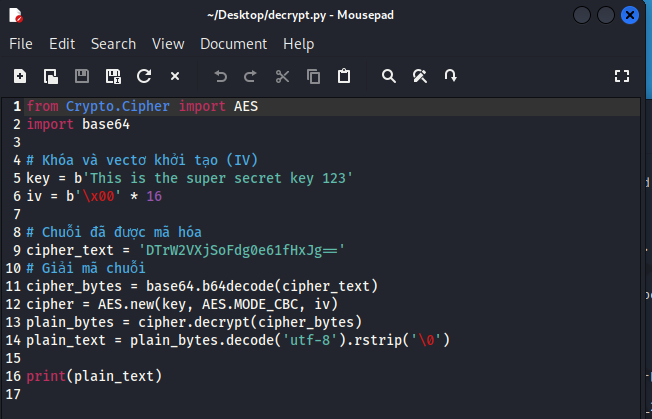
Yêu cầu 5 Viết chương trình giải mã đoạn dữ liệu mã hoá (python3 chẳng hạn…)



* Tại đường dẫn com/android/insecurebankv2/CryptoClass.class, chứa các hàm mã hoá và giả mã.
* Đây là đoạn mã java dùng thuật toán AES-256 với mode CBC và đệm PKCS5Padding để mã hóa
* Trong đây có key dùng để giải mã và cũng có iv4 cần thiết để có thể giải mã
* Sau khi được mã hóa nó được chuyển đổi sang chuỗi base64 và trả về kết quả



* Dựa vào đó attacker có thể giải mã được bằng đoạn code python sau

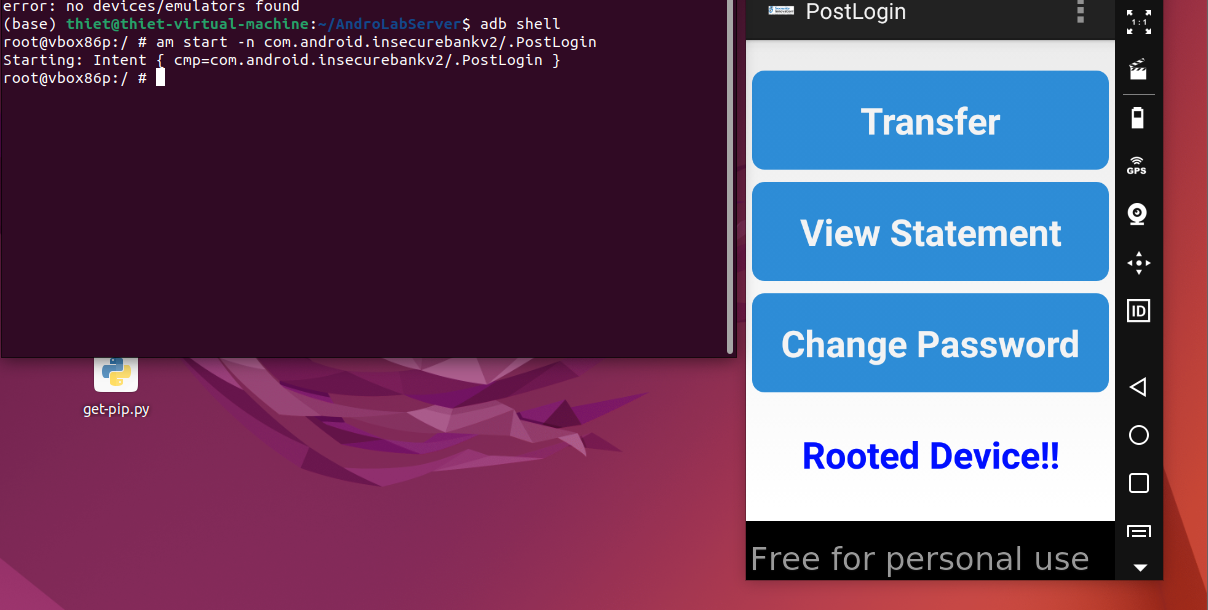


Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ

Mô tả được tạo tự động

* Đây là password của user “dinesh” sau khi đã được giải mã

Activity Hijacking



* Sau khi hijacking thì nó tự động nhảy sang tab PostLogin mà ta không cần đăng nhập

C.2 Dịch ngược (Reverse Engineering)

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Hệ điều hành

Mô tả được tạo tự động

* Sau khi ta decompiles thì ta vào đường dẫn Insecurebankv2/res/values/strings.xml
* Sửa value “is\_admin” thành yes

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ

Mô tả được tạo tự động

* Nhưng khi cài thì lại lỗi vì ta cần ký chứng thực trước khi cài đặt trên thiết bị

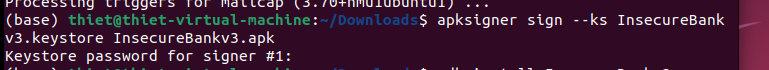
Ảnh có chứa văn bản, ảnh chụp màn hình

Mô tả được tạo tự động

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ

Mô tả được tạo tự động

* Sử dụng lệnh “keytool -genkey -v -keystore InsecureBankv3.keystore -alias InsecureBankv3 -keyalg RSA - keysize 2048 -validity 10000”



* Sử dụng công cụ apksigner

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ

Mô tả được tạo tự động

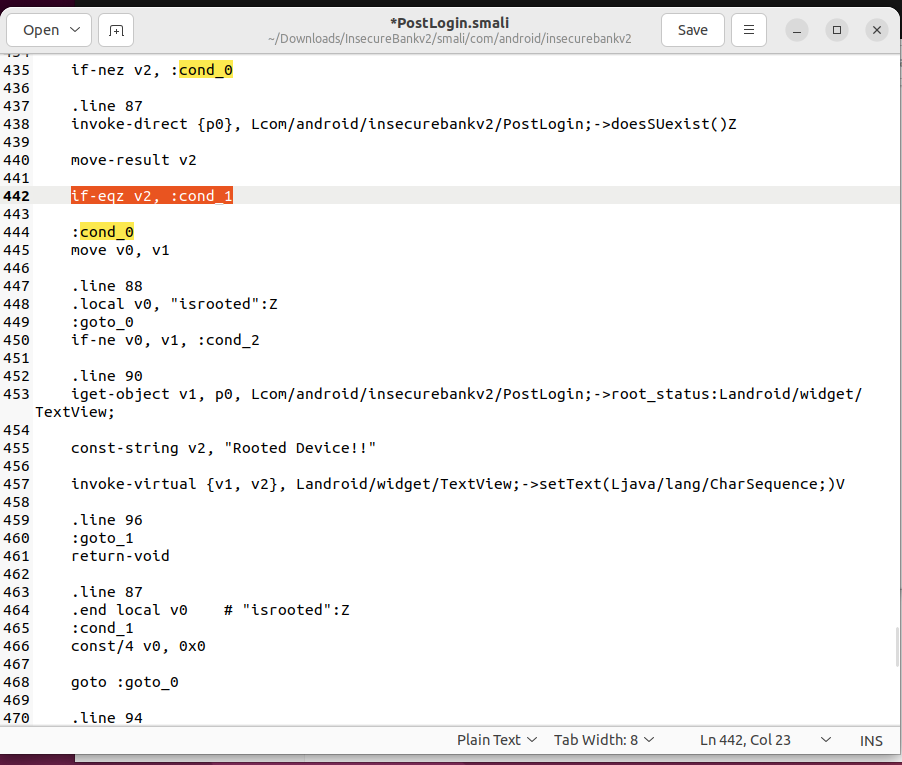
* Push lên thành công
* Ta phải xóa phiên bản cũ trước khi cài

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, đa phương tiện

Mô tả được tạo tự động

* Nút create user xuất hiện mà chỉ có admin mới có

Yêu cầu 6 Sinh viên điều chỉnh mã nguồn ứng dụng sao cho luôn hiển thị trạng thái “Rooted Device!!” với bất kỳ trạng thái nào của thiết bị.



* Đọc trong showRoot Status(), tại .line86 sẽ kiểm tra xem có tồn tại file /system/app/Superuser.apk hay không và gán kết quả vào v2, nếu = 1 nghĩa sẽ nhảy đến label :cond\_0 chạy dòng lệnh bên dưới label này để in ra dòng “isrooted”. Nếu v2 = 0 thì nghĩa là không tồn tại file Superuser.apk, thì nó sẽ kiểm tra tiếp tới doesSUexist(), kết quả gán vào v2, nếu như v2=0 thì nó sẽ nhảy đến cond\_1. Label này để in ra “not rooted”.
* Vậy để in ra “is rooted” trong mọi hoàn cảnh thì ta có thể xóa dòng được khoanh trong ô màu đỏ “if -eqz v2, :cond\_1” hoặc đổi lệnh đó thành “if -eqz v2, :cond\_0” để nó không nhảy đến cond\_1 mà chạy thẳng xuống dòng bên dưới là in ra “is rooted”.

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

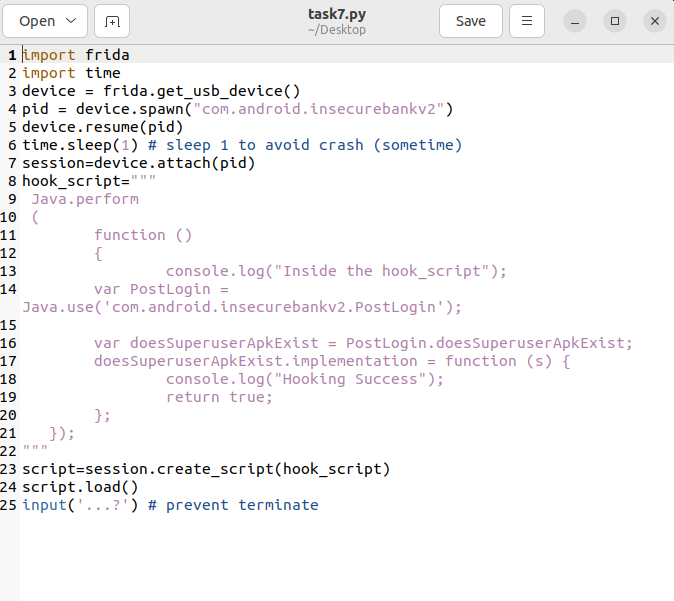
Mô tả được tạo tự động

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Phần mềm đa phương tiện

Mô tả được tạo tự động

* Kết quả sau khi sửa và compile rồi install trên máy

Yêu cầu 7 Hoàn thiện đoạn code trên và demo.



* để hook hàm doesSuperuserApkExist trong lớp PostLogin của ứng dụng Android "InsecureBankv2". Khi hàm này được gọi, nó sẽ in ra "Hooking Success" trên console và trả về giá trị true.

Ảnh có chứa văn bản, đồ điện tử, ảnh chụp màn hình, màn hình

Mô tả được tạo tự động

* ta cài frida-server trên shell của điện thoại sau đó cấp quyền và chạy

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ

Mô tả được tạo tự động

* Ta tiến hành chạy file python để hook
* Trên điện thoại lúc này sẽ bắt ta đăng nhập
* Khi đăng nhập thành công thì sẽ hiển thị Hooking Sccess

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Website

Mô tả được tạo tự động

* Kết quả khi đăng nhập thành công