한국사물인터넷학회논문지 Vol. 6, No. 4, pp. 65-72, 2020 https://doi.org/10.20465/KIOTS.2020.6.4.065

머신러닝 기반 블록체인 DApp 시스템 설계 및 구현

이형우[1*], 이한성[2]
[1]한신대학교 컴퓨터공학부 교수, [2]한신대학교 대학원 컴퓨터공학과 연구원

Design and Implementation of Machine Learning-based Blockchain DApp System

Hyung-Woo Lee[1*], HanSeong Lee[2]
[1]Professor, Div. of Computer Engineering, Hanshin University
[2]Researcher, Dept. of Computer Engineering, Hanshin University

요 약 본 논문에서는 지속적으로 급증하고 있는 안드로이드 악성 앱을 자동적으로 판별하기 위해 머신러닝 기법을 적용하여 프라이빗 블록체인을 토대로 웹 기반 DApp 시스템을 개발하였다. 공인 실험 데이터를 대상으로 안드로이드 악성 앱 판별에 96.2587% 정확도를 제공하는 최적의 머신러닝 모델을 선정하였고, 안드로이드 악성 앱에 대한 자동 판별 결과를 Hyperledger Fabric 블록체인 시스템 내에 자동적으로 기록/관리하였다. 또한 적법한 권한이 부여된 사용자만이 블록체인 시스템을 이용할 수 있도록 웹 기반의 DApp 시스템을 개발하였다. 따라서 본 논문에서 제시한 머신러닝 기반 안드로이드 악성 앱 판별 블록체인 DApp 시스템 개발을 통해 안드로이드 모바일 앱 이용 환경에서의 보안성을 더욱 향상시킬 수 있으며, 향후에 일반적인 범용 데이터를 대상으로 머신러닝과 블록체인을 결합한 보안 서비스로 발전시킬 수 있을 것으로 기대된다.

주제어 : 머신러닝, 블록체인, DApp 시스템, 안드로이드, 악성 앱 탐지, 모바일 보안

Abstract In this paper, we developed a web-based DApp system based on a private blockchain by applying machine learning techniques to automatically identify Android malicious apps that are continuously increasing rapidly. The optimal machine learning model that provides 96.2587% accuracy for Android malicious app identification was selected to the authorized experimental data, and automatic identification results for Android malicious apps were recorded/managed in the Hyperledger Fabric blockchain system. In addition, a web-based DApp system was developed so that users who have been granted the proper authority can use the blockchain system. Therefore, it is possible to further improve the security in the Android mobile app usage environment through the development of the machine learning-based Android malicious app identification block chain DApp system presented. In the future, it is expected to be able to develop enhanced security services that combine machine learning and blockchain for general-purpose data.

Key Words : Machine Learning, Blockchain, DApp System, Android System, Malicious App Detection, Mobile Security.

이 논문은 2020학년도 한신대학교 학술연구비의 지원에 의하여 연구되었음.
*교신저자 : 이형우(hwlee@hs.ac.kr)
접수일 2020년 10월 14일 수정일 2020년 11월 27일 심사완료일 2020년 11월 30일

1. 서론

기존의 악성코드 탐지 규칙만으로는 급격히 고도화되고 지능화되는 신종·변종 악성코드를 효율적으로 탐지·검출하기에는 어려움이 있다. 이에 알려진 악성코드를 빅데이터 셋으로 구축한 한 후 이를 토대로 머신러닝 기법[1,2]을 적용하여 자동화 기반 검출 성능을 향상시키기 위한 연구가 활발히 진행되고 있다. 카스퍼스키 랩(Kasperskey Lab) 등에서 제시한 머신러닝 기반 악성코드 검출 방법[3]인 경우 분석 대상 파일에 대한 특징정보를 추출한 후 머신러닝 기법을 적용하여 탐지 성능 향상과 더불어 오탐율을 낮출 수 있는 방법 등에 대한 연구를 진행하고 있으므로 해당 기법은 안드로이드 모바일 앱에 대한 정상/악성 여부를 판별하는 과정에도 그대로 적용 가능하다. 하지만 일반 사용자가 자신의 모바일 단말에 임의의 앱을 설치할 경우에 해당 앱이 정상인지 악성인지 판별하기 위해서 현재까지는 대부분 각 사용자별로 본인의 단말내에 모바일 백신 등을 설치하여 사용자마다 개별적으로 검사하는 과정을 수행하고 있어서 이를 효율적으로 개선할 수 있는 방법이 필요한 시점이다. 만일 특정 모바일 앱의 정상/악성 여부를 신뢰할만한 주체가 미리 검사한 후 판별 결과를 누구나 검색 가능하고 신뢰할 수 있는 검증된 시스템 내에 저장/관리한다면 결과적으로는 모바일 앱에 대한 정상/악성 판별 과정의 신뢰도를 향상시키고 또한 악성코드 검사 과정의 효율성을 증대할 수 있을 것으로 예상된다.

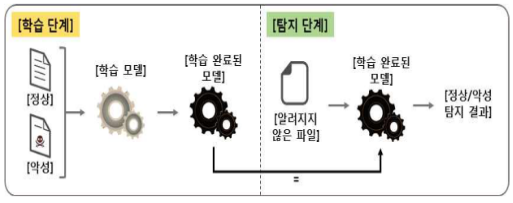
이에 본 연구에서는 최적의 머신러닝 기법을 적용하여 정상/악성 앱 판별의 정확도를 향상시키고, 각 앱 별 판별된 결과는 특정 권한을 소유한 주체(예: 공공기관/지자체 또는 신뢰할만한 기관 등)만이 생성/저장/관리하도록 프라이빗 블록체인 시스템[4]을 적용하여 구현하였다. 그리고 프라이빗 블록체인 시스템 내부에 저장된 정상/악성 판별 결과는 일반 사용자 누구든지 검색 가능하도록 하기 위해 웹 인터페이스로 손쉽게 접근할 수 있도록 블록체인 DApp을 설계 및 구현하였다.

2장에서는 관련 연구 내용을 제시하고, 3장 및 4장에서는 머신러닝 기반 안드로이드 악성 앱 판별 블록체인 시스템 모델과 구현 결과를 제시하고, 5장에서는 악성 앱 판별 기능을 제공하는 블록체인 기반 DApp 구현 결과 및 성능 비교 평가 과정을 수행하였다.

2. 관련 연구

2.1 머신러닝 기반 악성코드 탐지

머신러닝[1-3] 기반의 악성코드 탐지란 정상파일 및 악성코드가 포함된 파일(악성코드)을 머신러닝 학습 모델을 이용하여 학습시킨 후 학습된 머신러닝 모델을 이용하여 의심스러운 파일의 정상/악성 여부를 자동 탐지하는 방법을 의미한다. 학습 단계에서는 머신러닝 학습 모델을 구축하는 단계로, 파일들의 특징정보(문자열, 명령어, 바이트 정보, API 호출 정보 등)와 레이블(정상/악성 여부)로 학습시켜 머신러닝 모델이 악성코드 탐지에 최적화 되도록 설정한다. 탐지 단계에서는 학습된 모델을 이용하여 입력된 파일이 정상인지 악성인지 구별하는 과정을 수행하게 된다[4-7].



[Fig. 1] Normal/malignant code detection using machine learning

2.2 카스퍼스키 랩 악성코드 탐지

카스퍼스키 랩은 악성코드 탐지를 위해 여러 단계로 구성하여 각 단계별로 머신러닝을 활용한 파일 학습 또는 악성 코드 탐지 과정을 수행토록 하였다. 우선 시스템 구성도를 살펴보면 악성코드 탐지를 위해 Pre-execution 단계와 Post-execution 단계로 구분하여, 실행 전(Pre-execution) 단계에서는 악성코드 실행 없이 수집 가능한 정보(파일 포맷, 바이트, 추출된 텍스트 등)를 대상으로 머신러닝 모델에서 학습 과정을 수행하여 악성코드를 탐지하는 과정을 수행한다. 그리고 실행 파일의 기본적인 특징정보(Lightweight Features)로 유사성 해시(Similarity Hash) 함수를 학습시킨 후 정상 파일과 악성 코드를 분류하는 과정을 수행한다. 유사성 해시 함수는 특징정보의 해시 값에 따라 악성코드를 적절한 버킷으로 분류하는 과정을 제공하며, 만일 버킷이 정상파일과 악성코드가 혼재할 경우 심화 탐지 과정을 수행한다. 심화 탐지 과정에서는 실행 파일에서 추출 가능한 모든 특징정보(Heavy Features)를 이용하여 유사성 해시 함수로 학습시킨 후 앙상블 알고리즘을 이용하여 정상파일과 악성파일을 분류하는 과정을 수행토록 하였다. 만일 분류 결과에 정상파일과 악성코드가 혼재할 경우 딥러닝 기반

회귀 악성코드 학습 단계를 수행토록 하였다[4,5].

만일 암호화, 난독화 등으로 Pre-execution 과정으로 악성코드를 구별하기 어려운 경우에는 Post-execution 과정을 수행한다. 악성코드를 실행하여 수집되는 정보(발생한 이벤트, 프로세스 행위 기록 등)를 이용하여 딥러닝 모델 기반 학습 과정을 수행하는 방식이다. 따라서 악성코드 실행 시 수집되는 프로세스의 동작 기록과 행동 그래프 및 행동 패턴 등과 같은 특징정보를 토대로 정상파일과 악성코드를 구별하는 방식을 제공한다[8-10]. 실험결과 카스퍼스키 랩은 강건성, 확장성, 처리율 등에서 장점을 보이며 악성코드의 작은 변화에도 탐지 성능이 향상되는 특징을 보이며, 대량의 악성코드 처리에 적합하다는 장점이 있다. 하지만, 카스퍼스키 랩을 통해 구별된 정상파일과 악성코드 정보에 대해서는 신뢰할만한 저장/기록 체계가 구축될 필요가 있어 본 연구에서는 블록체인 시스템 등을 이용하여 판별 결과를 관리하는 방법을 적용하였다.

2.3 블록체인 기반의 DApp

블록체인 DApp(Decentralized Application)은 기존의 중앙화된 어플리케이션 동작과 달이 블록체인 플랫폼을 기반으로 동작하는 분산형 어플리케이션으로 블록체인 시스템에서 구동하는 응용 프로그램, 프로그램 툴을 모두 통칭하는 것이다. 기존 구글플레이 또는 애플 앱스토어에서 쉽게 다운로드 받을 수 있는 기본 어플리케이션과 달리 DApp은 블록체인 시스템 내에서 사용하기 위해 제작된 차별화된 어플리케이션을 의미한다. 따라서 본 연구에서는 머신러닝 학습 모델을 적용하여 정상파일과 악성코드를 자동으로 구별한 후 이를 블록체인 시스템 내에 저장/관리하는 DApp을 개발하여 임의의 안드로이드 모바일 앱에 대해 정상/악성 여부를 자동으로 판별할 수 있는 시스템을 구현하였다.

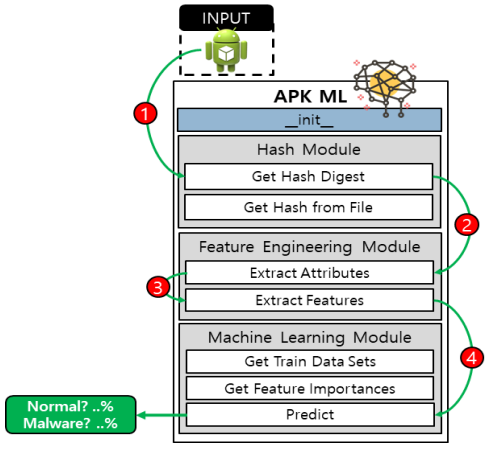
3. 머신러닝 기반 악성 앱 판별 블록체인 시스템 모델

3.1 머신러닝 기반 안드로이드 악성 앱 판별

우선 안드로이드 악성 앱 판별에 가장 좋은 성능을 보인 머신러닝 모델을 채택하여 안드로이드 악성 앱 판별 시스템을 구현하였다. 실험결과 F1 Scorea를 토대로 탐지 성능이 가장 좋은 Random Forest 모델을 이용하였으며, 임의의 파일에 대해 정상/악성코드를 판별할 수 있도록 머신러닝 기반 APK 탐지 모듈인 “APK ML”을 파이썬으로 구현하였다.

APK ML은 파이썬 언어로 구현하였으며, 객체가 생성되면 생성자 __init__ 메소드를 통해 머신러닝 모델을 초기화한다. 그리고 머신러닝 모델에 데이터를 학습시켜 악성 앱 판별을 할 수 있도록 준비한다. APK ML은 안드로이드 APK 파일을 입력 받아 다음과 같이 4 단계의 악성 앱 판별 과정을 수행한다.

① APK 파일의 해시값을 계산한다.
② APK 파일에서 속성을 추출하여 JSON 형태로 변환한다.
③ 속성 중에서 특징을 추출하여 CSV 형태로 변환한다.
④ Predict 메소드에서는 머신러닝 모델에 해당 특징이 정상과 악성코드 중 어디에 가까운지 측정한다. 마지막으로 Predict 메소드를 통해 머신러닝이 판단한 정상과 악성 앱 확률이 몇 퍼센트 인지 반환한다.



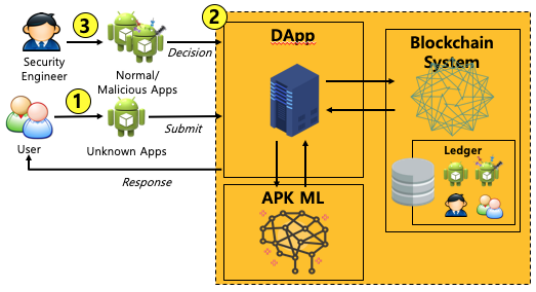
[Fig. 2] Machine learning-based Android malicious app identification process

3.2 블록체인 기반 시스템 모델

본 논문에서는 안드로이드 애플리케이션에서 추출한 속성을 저장할 수 있는 블록체인 시스템을 구현하였다. 그리고 안드로이드 앱 블록체인 시스템과 정상과 악성 애플리케이션의 판별하는 APK ML 시스템을 통합한 블록체인 기반의 안드로이드 악성 앱 자동 판별 시스템을 개발하였다.

시스템의 작동 과정은 다음과 같다. ① 일반 사용자가

안드로이드 애플리케이션 분석 또는 악성 여부 판별을 제안 시스템에 요청한다. 그러면 ② 제안 시스템은 요청받은 안드로이드 애플리케이션의 속성과 특징을 추출한다. 그리고 추출된 특징으로 머신러닝 모델에 악성 여부를 자동 판별한다. 머신러닝으로부터 판별된 악성 여부는 일반 사용자와 보안 전문가 모두 확인할 수 있다. ③ 보안 전문가는 정상 또는 악성 애플리케이션의 판별 결과를 확인하여 악성 여부를 블록체인 시스템에 등록한다[13].

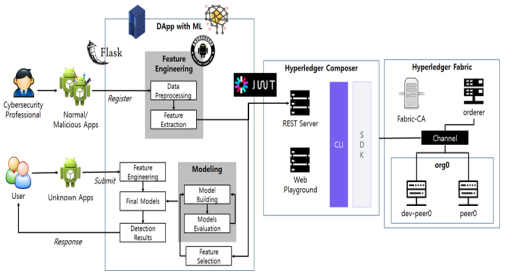


[Fig. 3] The proposed machine learning-based blockchain DApp system operation structure

4. 안드로이드 앱 정상/악성 판별 블록체인 시스템 구현

4.1 안드로이드 앱 블록체인 시스템 구조

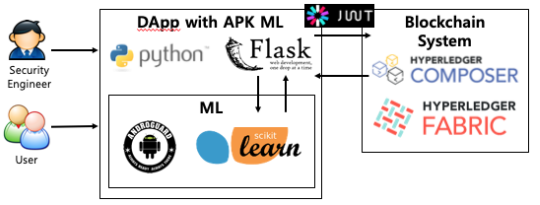
머신러닝이 적용된 블록체인 기반의 안드로이드 악성 애플리케이션 판별 시스템 구조는 다음 그림과 같다. 제안 시스템은 DApp with APK ML, Hyperledger Composer, Hyperledger Fabric 3가지로 구성된다. DApp with APK ML은 파이썬 기반으로 구현하였다. 사용자는 DApp with APK ML을 통해 머신러닝과 블록체인 서비스를 받는다. 머신러닝 서비스는 특징 공학과 정상과 악성 앱을 판별하기 위한 머신러닝이 수행된다. 블록체인 서비스는 Hyperledger Fabric과 Hyperledger Composer로 안드로이드 앱 정보를 분산원장에 관리한다. DApp with APK ML과 블록체인 서비스의 실행 환경이 서로 달라 Hyperledger Composer에서 제공하는 REST 서버를 이용하여 해결하였다[14].



[Fig. 4] Detailed structure diagram of the proposed system

4.2 안드로이드 앱 블록체인 시스템의 적용 기술

제안 시스템에 적용된 기술은 APK ML, 블록체인, DApp 시스템으로 구분된다. APK ML 시스템은 Androguard[9] 라이브러리로 안드로이드 애플리케이션 분석 및 특징 공학을 수행하고 scikit-learn 라이브러리로 머신러닝 모델 구축 및 구현한다. 블록체인 시스템은 Hyperledger Fabric으로 권한형 프라이빗 블록체인 네트워크를 구성한다. 블록체인 네트워크를 빠르게 구성할 수 있게 도와주는 도구로 Hyperledger Composer를 이용해 안드로이드 애플리케이션 정보와 사용자 정보를 저장할 수 있는 안드로이드 앱 블록체인 네트워크를 구성한다. APK ML, 블록체인 시스템이 구성되면 파이썬 웹 프레임워크인 Flask로 DApp 웹 서버를 구현하고 DApp 웹 서버에 로그인한 사용자 세션을 블록체인 시스템까지 유지할 수 있도록 JSON 웹 토큰인 JWT(JSON Web Token)을 적용하였다.

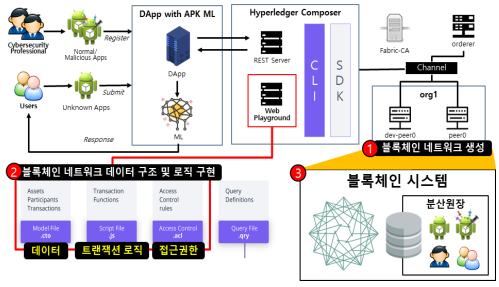


사용 기술	설명	용도
Androguard	파이썬 안드로이드 앱 정적분석 도구	머신러닝에 필요한 특징공학 구현
scikit-learn	파이썬 머신러닝 라이브러리	머신러닝 모델 구축 및 구현
Hyperledger Fabric	권한형 프라이빗 블록체인 프레임워크	블록체인 네트워크 개발
Hyperledger Composer	블록체인 네트워크 및 애플리케이션 개발도구	
JWT	JSON 웹 토큰	DApp에 로그인한 사용자 세션을 블록체인 시스템까지 유지하기 위한 토큰
Flask	파이썬 웹 프레임워크	웹 기반 DApp 구현

[Fig. 5] Detailed system operation flow and list of tools [11,12] used

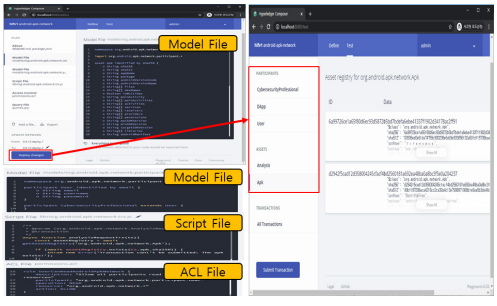
4.3 안드로이드 앱 블록체인 시스템 구현 과정

안드로이드 앱 블록체인 시스템을 구현하기 위한 과정은 다음 그림과 같다. 첫 번째, 블록체인 네트워크를 생성한다. 두 번째, 블록체인 네트워크에 데이터 구조 및 로직을 구현한다. 마지막으로 구현 결과를 블록체인 네트워크에 반영하면 안드로이드 앱 정보를 관리할 수 있는 분산원장이 생성된다[15].



[Fig. 6] The process of storing Android malicious app identification results in the blockchain

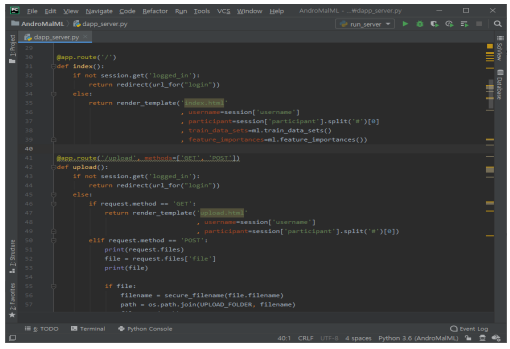
다음 그림은 Hyperledger Composer의 Web Playground를 이용하여 안드로이드 앱 블록체인 시스템의 데이터 구조 및 로직을 구현한 것이다. 분산원장에 저장될 안드로이드 앱, 사용자 정보 등의 데이터에 대한 구조를 Model File에 구현하였다. 블록체인 시스템에 데이터를 생성할 수 있도록 Script File에 트랜잭션 로직을 구현하였다. ACL File은 사용자와 보안 전문가가 블록체인 시스템에 접근할 수 있는 권한을 설정하였다.



[Fig. 7] Detailed storage information in the blockchain

5. 머신러닝 기반 DApp 구현 결과

DApp은 Flask 웹 프레임워크를 이용하여 구현하였으며 사용자와 보안 전문가는 웹을 통해 DApp 시스템에 접근할 수 있다. 그리고 APK ML과 안드로이드 앱 블록체인 시스템을 연계한 DApp with APK ML을 다음 그림과 같이 구현하였다.



[Fig. 8] DApp implementation result (1)

5.1 사용자 로그인 및 등록

DApp with APK ML은 사용자 로그인 및 등록을 할 수 있다. 사용자 등록에서 일반 사용자 또는 보안 전문가를 선택할 수 있다. 그리고 블록체인 네트워크에 사용자 정보가 등록된다. 다음 그림은 DApp에서 User1 사용자를 등록하고, 블록체인 시스템에서 사용자 등록 트랜잭션을 확인한 것이다.

[Fig. 9] DApp implementation result (2)

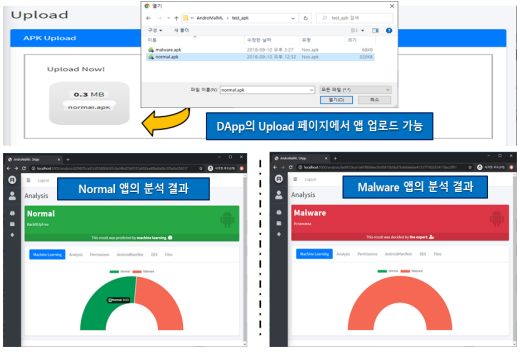
메인 페이지는 DApp에 동작 중인 머신러닝 정보를 보여준다. 메인 페이지의 좌측 패널에 로그인한 사용자 정보와 메뉴를 보여준다. 로그인 정보에서 사용자와 보안 전문가에 따라 표시되는 아이콘이 다르게 하였다. 페이지 상단은 학습 데이터 중 정상과 악성의 개수를 보여준다.

페이지 하단에는 랜덤 포레스트에서 추출한 특징 중요도(feature importance)에서 상위 20개를 그래프를 보여준다.

[Fig. 10] DApp implementation result (3)

5.2 머신러닝 기반 안드로이드 악성 앱 자동 판별

Upload 페이지에서는 APK 파일을 DApp with APK ML에 업로드하면 정상과 악성 앱 판별을 할 수 있다. 다음 그림은 사용자로 로그인하여 APK 파일을 업로드하는 과정이다. 안드로이드 앱이 서버에 업로드되면 APK ML 시스템에서 이를 분석하여 결과를 사용자에게 Analysis 페이지로 보여준다. Analysis 페이지에서 머신러닝이 자동 판별한 안드로이드 앱의 정상 또는 악성 여부를 확인할 수 있다.



[Fig. 11] Android malicious app automatic detection result (1)

그리고 안드로이드 앱을 분석한 정보 및 메타데이터도 확인할 수 있도록 6개의 탭을 배치하였다. Machine Learning 탭은 그래프로 머신러닝이 정상과 악성을 판별한 결과를 볼 수 있었다. Analysis 탭은 안드로이드 앱의 해시 값, 패키지 명, 버전 정보를 확인할 수 있다. Permissions 탭은 안드로이드 앱에서 사용 중인 퍼미션 목록을 보여준다. AndroidManifest 탭에서는 안드로이드 앱에 있는 AndroidManifest.xml 파일을 확인할 수 있다. DEX 탭은 안드로이드 앱의 DEX 파일의 Multi DEX 여부, ASCII 난독화 여부 등을 보여준다. Files 탭에서는 안드로이드 앱 내에 존재하는 파일 목록을 확인할 수 있다.

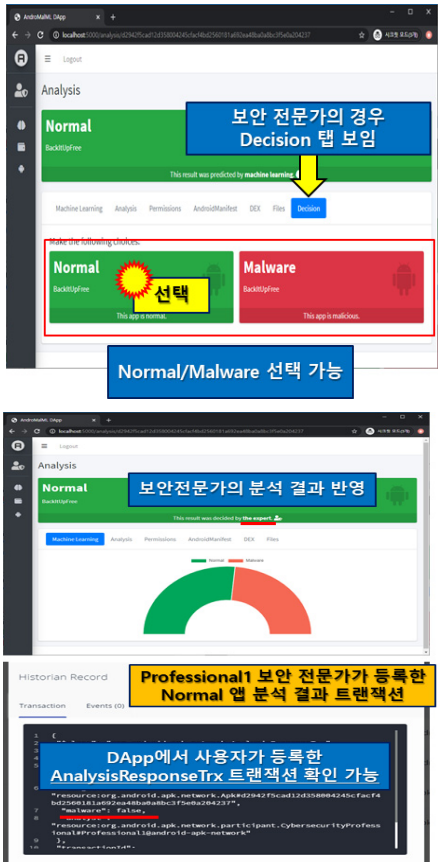
[Fig. 12] Android malicious app automatic detection result (2)

5.3 보안전문가용 안드로이드 앱 분석

보안전문가는 안드로이드 앱을 DApp with APK ML

에 업로드하여 Anlaysis 페이지에서 안드로이드 앱 분석과 머신러닝 결과를 확인한다. Analysis 페이지에서 Decision 탭은 보안 전문가에게만 보이는 기능이다. 보안 전문가는 분석 정보를 종합하여 Deicision 탭을 통해 해당 앱이 정상인지 악성인지를 선택하여 시스템에 반영하게 된다.

다음 그림에서 등록한 안드로이드 앱을 Normal로 선택하였다. 정상과 악성 여부를 선택하면 페이지가 업데이트된다. 그러면 페이지 상단에 표시되어 있던 내용이 머신러닝 결과에서 보안 전문가의 선택 결과로 변경된다. 그리고 Hyperledger Composer에서 보안 전문가가 등록한 Decision 결과에 대한 트랜잭션을 확인할 수 있다.



[Fig. 13] Android malicious app automatic detection result (3)

5.4 기존 연구와의 비교 분석 및 성능 평가

기존의 안드로이드 악성 앱 탐지 시스템과 본 논문에서 제시한 시스템을 비교한 결과가 다음 표 1과 같다. 3,154개의 안드로이드 앱을 제안 시스템에 정상/악성 여부를 자동 판별한 결과 96.2587% 정확도를 보였다. 이는 기존 연구보다 더 높은 정확도를 가진다. 또한, 머신러닝과 블록체인 시스템을 연계한 정상과 악성 앱 판별 시스템이 없는 것을 알 수 있다.

〈Table 1〉 Android Malware Detection System Comparison

	Pattern DB	Bitmap DEX	Proposed System
Platform	Android	Android	Android
Target	Class/ Method Name	DEX file	APK file
Algorithm	KMP	CNN	Random Forest
Detection Rate	88%	84.73% ~ 87.96%	96.2587%
Data Management	Database	-	Blockchain

6. 결론

본 논문에서는 최근 급증하고 있는 안드로이드 악성 앱을 높은 정확도로 자동적으로 판별하기 위해 최적의 성능을 제공하는 머신러닝 기법을 적용하여 프라이빗 블록체인 기반 DApp을 개발하였다. 구체적으로 공인 실험 데이터를 대상으로 가장 높은 판별 성능을 제공하는 머신러닝 모델을 도출하였으며, 안드로이드 악성 앱에 대한 자동 판별 결과를 Hyperledger Fabric 블록체인 시스템 내에 저장/관리하고 이를 적법한 권한이 부여된 사용자와 보안 관리자가 손쉽게 블록체인 시스템을 접근/이용할 수 있도록 웹 기반의 DApp 시스템을 개발하였다. 본 시스템 개발을 통해 안드로이드 모바일 앱 이용 환경에서의 보안성을 더욱 향상시킬 수 있으며, 앞으로도 일반적인 데이터를 대상으로 머신러닝과 블록체인을 결합한 서비스를 제공할 수 있을 것으로 기대된다.

REFERENCES

[1] SAS Institute, “Machine Learning: What it is & why it matters”, https://www.sas.com/en_us/insights/analytics/machine-learning.html (November 1, 2020)

[2] H. Lee, S. Chung, E. Choi, “A Case Study on Machine Learning Applications and Performance Improvement in Learning Algorithm,” Journal of Digital Convergence, Vol.14, No.2 pp.245-258, 2016.

[3] Machine Learning for Malware Detection, Kasperskey Lab, Edward Raff, Malware Detection by Eating a Whole EXE, NVIDIA, 2017.

[4] The Linux Foundation Project, Hyperledger Project URL: https://www.hyperledger.org

[5] Security Technology Research Team, “Introduction of malicious code detection research by foreign companies using machine learning,” Financial Security Institute Technical Report, 2018-029, 2018, 4, http://www.fsec.or.kr/common/proc/fsec/bbs/42/file DownLoad/1520.do

[6] Dong-Hyeok Park, Eui-Jung Myeong, Joobeom Yun, "Efficient Detection of Android Mutant Malwares Using the DEX file", Korea Institute Of Information Security And Cryptology, Vol.26, No.4, pp.895-902. 2016.

[7] Da-Hye Kim, Myeon-ggeon Lee, Min-Su Song, Seong-je Cho, "Machine Learning based Android Malware Detection using Gray Scale Images", KOREA INFORMATION SCIENCE SOCIETY, 1245-1247. 2018.

[8] S.J Cho, “Latest Android malicious app trends and detection techniques,” iitp technical report, 2019, 9

[9] Symantec. Internet Security Threat Report. Volume 23. March 2018. https://docs.broadcom.com/doc/istr-23-2018-en.

[10] Victor Chebyshev. Mobile malware evolution 2019. February 25, 2020. http://securelist.com/mobile-malware-evolution-2019 /96280/.

[11] Androguard. https://github.com/androguard/androguard.

[12] scikit-learn. https://scikit-learn.org/.

[13] S.M.Hwang and H.W.Lee, "Identification of Counterfeit Android Malware Apps using Hyperledger Fabric Blockchain," Journal of Internet Computing and Services, Vol.20, No.2, pp.61-68, 2019. DOI: 10.7472/jksii.2019.20.2.61.

[14] H.S.Lee and H.W.Lee, "Consortium Blockchain based Forgery Android APK Discrimination DApp using Hyperledger Composer," Journal of Internet Computing and Services, Vol.20, No.5, pp.9-18, 2019. DOI: 10.7472/jksii.2019.20.5.9.

[15] H.S.Lee and H.W.Lee, "Optimal Machine Learning Model for Detecting Normal and Malicious Android Apps,“ Journal of Korea Internet on Things Society, Vol.6, No.2, pp.1-10, 2020. DOI: 10.20465/KIOTS.2020.6.2.001.

이 형 우(Hyung-Woo Lee) [종신회원]

- 1994년 2월 : 고려대학교 컴퓨터학과 (학사)
- 1996년 2월 : 고려대학교 컴퓨터학과 (석사)
- 1999년 2월 : 고려대학교 컴퓨터학과 (박사)
- 1999년 3월 ~ 2003년 2월 : 백석대학교 정보통신학부 교수
- 2003년 3월 ~ 현재 : 한신대학교 컴퓨터공학부 교수

〈관심분야〉
사물인터넷, 정보보호, 모바일 보안 및 디지털 포렌식

이 한 성(Hanseong Lee) [정회원]

- 2017년 2월: 한신대학교 컴퓨터공학부 졸업(학사)
- 2018년 9월~2020년 8월: 한신대학교 일반대학원 컴퓨터공학과 (석사)

〈관심분야〉
블록체인 기술, 모바일 보안, 디지털포렌식, 리버스 엔지니어링 등