코드 분리 기반 스마트 자동차 소프트웨어 보안 시스템*

김경리, 서혜민, 태희지, 이수원**
숭실대학교 소프트웨어학부
e-mail:kkr0373@naver.com

A Smart Car Software Security System based on Code Splitting

Kyung-ri Kim, Hyemin Seo, Heejee Tae, Soowon Lee
School of Software, Soongsil University

요 약

최근 스마트 카가 발전하면서 물리적 키에서 스마트 키로 차량용 키도 변화하고 있다. 이에 따라 스마트 키에 대한 해킹 위협도 증가하고 있으며 스마트 키의 보안은 점점 중요해지고 있다. 기존의 스마트 키는 자동차 시스템이 부팅 된 후에 데이터 값을 비교하는 방식으로 사용자 인증을 진행한다. 이러한 방식은 시스템이 이미 부팅된 상태이므로 여전히 해킹의 위협이 존재한다는 점에서 안전하지 않다. 본 연구에서는 이를 개선하기 위해 Python 코드 분리 기술과 APK 바이트 코드 분리 기술을 자동차 부팅 코드와 엔진 APK의 바이트 코드에 적용하는 방법을 제안한다. 제안 기술이 적용되지 않은 APK와 적용된 APK를 리패키징하여 해킹에 대한 본 연구의 안정성을 검증하였다

1. 서론

최근 자동차 키가 스마트 키로 진화하면서 키 복제나 리패키징과 같은 보안 위협이 증가하고 있다. 코드 분리 기술은 코드의 핵심코드를 사용자 인증키로 사용하는 기술이다. 전체 코드 중 사용자가 원하는 부분의 코드를 선택하여 분리할 수 있으므로, 분리되는 핵심코드는 사용자 개개인의 고유한 키 역할을 한다. 따라서 핵심코드가 재조립되지 않으면 전체 시스템을 부팅할 수 없고, 사용자의 디바이스가 있고 인증이 성공했을 때만 코드 재조립을 통해 시스템이 부팅된다. 본 연구에서는 기존의 스마트 키의 해킹 위협성에 대응하기 위해 Python 코드 분리 기술과 APK 바이트 코드 분리 기술을 통한 스마트 카 보안 시스템을 제안한다.

2. 관련 연구

[1]은 자바 소스코드 분리 기술을 활용하여 모바일 코드를 보호하는 방법을 제안하였다. 하지만 이 방법은 JAVA코드에 한정되어 있고, 모바일 코드에만 적용해 그 결과를 검증했다는 점에서 한계점을 지닌다. [2]는 사용자 개인 정보를 보유하고 있는 애플리케이션 안에 악성행위를 통제할 수 있도록 모니터링 기능을 삽입을 제안하였다. 하지만 이러한 방식은 사용자가 애플리케이션 코드를 수정하는 데 어려움이 있다는 점에서 한계가 있다. [3]은 양방향 통신을 이용한 차량용 스마트 키 시스템을 제안하였다. 이 방법은 운전자가 키를 갖고 있는 것만으로 자동차의 작동이 가능한 Passive Entry 기술이다. 이는 기존 터치 방식에서 발전된 형태이기는 하나, 데이터 값으로 인증하므로 부팅 후 보안위협이 있다는 점에서 한계점을 가진다. [4]에서 제시된 방법은 스마트폰의 보안을 지킬 수 있는 별도의 애플리케이션을 설치하는 방식으로, 보통 Launcher 애플리케이션을 통한 보호 기법들이 사용되는 추세이다.

3. 제안 방법

(그림 1) 코드 분리

코드 분리 기술이란 (그림 1)과 같이 본래의 소스 코드에서 핵심이 되는 코드 일부분을 분리하여 사용자 디바이스에 별도 저장하는 기술이다. 실행 시에 분리된 코드와 나머지 부분을 재조립을 통해 원래의 코드로 복원하여 사용한다. 본 연구에서는 이러한 코드 분리 기술을 Secure Booting과 Secure Infotainment에 적용하였다.

* 본 논문은 서울어코드활성화지원사업에서 지원되었음
** 교신저자임

3.1 Secure Booting

(그림 2) Secure Booting

(그림 2)는 Secure Booting의 작동 과정이다. 코드 분리 모듈을 통해 자동차의 부팅에 관여하는 ROS Core를 Core Code와 Remaining Code로 분리한다. 이때, Core Code가 사용자 고유의 키 역할을 하며, Core Code를 직접 지정할 수 있기 때문에 타 사용자와 키가 중복될 위험에서 벗어날 수 있다. 이러한 방식으로 분리된 Core Code는 사용자 디바이스에, 분리되고 남아 있는 Remaining Code는 스마트 카에 저장한다. 이후 자동차의 부팅 시에 사용자 디바이스의 앱을 통해서 사용자 인증 과정을 거친다. 성공적으로 인증이 완료되면 Core Code가 스마트 카로 전송되어 Core Code와 Remaining Code의 재조립이 일어나게 되면 본래의 Code가 완성되어 자동차가 안전하게 부팅된다.

3.2 Secure Infotainment

(그림 3) Secure Infotainment

(그림 3)은 Secure Infotainment System의 작동과정이다. 자동차 부팅 이후 동작하는 Infotainment System에 대한 보안 역시 필요하다. 코드 분리 모듈을 통해 Infotainment System의 애플리케이션을 Core Code와 Remaining Code로 분리한다. 이러한 방식으로 분리된 Core Code는 사용자 디바이스에, 분리되고 남아있는 Remaining Code는 Infotainment System 상에 저장한다. 이후 애플리케이션 동작 시에 사용자 디바이스로부터 Core Code를 수신 받아 Remaining Code와 재조립한다. 따라서 Core Code 없이는 앱이 작동되지 않기 때문에 앱의 안전성을 보장할 수 있다.

4. 실험 및 평가

제안 시스템의 안정성을 검증하기 위해서 코드 분리 기술을 활용하여 리패키징 공격 방어가 가능한지에 관한 실험을 진행하였다. 리패키징이란 앱의 제작 과정을 거꾸로 하여 앱의 최초 형태인 소스 코드로 변환한 후 소스 코드를 수정하거나 다른 코드를 삽입하고 앱을 다시 제작하는 일련의 과정이다. 또한, 환경을 동일하게 맞추기 위하여 같은 인포테인먼트 안에 있는 Engine Start 앱에 적용하여 검증하였다. 인포테인먼트 시스템 상에 존재하는 Engine Start 앱은 단순히 엔진을 켜거나 끌 수 있는 기능을 가진 앱이다. 코드 분리 기술을 적용하지 않았을 때와 적용하였을 때 리패키징 가능 여부를 기준으로 실제 해킹 위협에 대해 대응할 수 있는지 판단하였다.

코드 분리 기술을 적용하지 않은 기존 시스템에 리패키징 기술을 적용하여 앱이 변조 가능한지 실험하였다. 이 앱을 리패키징 기술로 변조하여 엔진을 켜거나 끌 때, 자동차의 Speed값을 변조하는 부분을 추가하여 악성 앱으로 변조하였다. 이때, 악성 앱이 정상적으로 작동되는 것을 확인하였고 악성 앱의 Speed값 조작에 따라서 계기판의 Speed값이 변경되었다.

기존 인포테인먼트의 앱에 코드 분리 기술을 적용하여 핵심 코드를 분리한 후 리패키징 기술을 적용하여 앱의 변조가 가능한지 실험하였다. 코드 분리 기술을 사용한 제안 시스템에서는 앱의 중요 코드를 외부로 분리함으로써 시스템의 코드를 알 수 없어 리패키징이 불가능하였다. 따라서 이를 통해 코드분리 기술을 사용하였을 때, 중요 코드의 노출을 원천 차단할 수 있고 리패키징으로부터 핵심 로직을 안전하게 보호할 수 있음을 확인할 수 있었다.

5. 결론 및 향후 연구

논문에서는 코드분리 기술을 개발하여 기존 스마트 키 방식에서 더 강력한 보안 기능을 제공하는 방식을 제안하였다. 기존의 스마트 키 방식은 단순한 데이터 값으로 사용자를 인증하기 때문에 부팅 후 위협에 대해 취약하다. 그러나 제안된 방식은 부팅에 관여하는 코드를 분리하여 사용자 인증을 수행하는 방식이므로 부팅 후 위협을 원천 차단하였다. 따라서 제안된 방식은 실험을 통해 기존의 위협들에 대해 보다 강력한 보안이 가능함을 확인하였다.

향후 연구방향으로는 좀 더 효과적인 사용자 편의성을 위하여 새로운 기능을 추가하는 것과 코드를 저장해 둘 사용자 디바이스를 웨어러블 기기 등으로 확대하는 것 등이다. 또한 실제 자동차 시스템에 적용하여 안정성을 검증하는 것도 필요하다.

참고문헌

[1] 박태용, "자바 소스코드 분리 기반 모바일코드 보호 기법", 숭실대학교 대학원. 2017.

[2] R. Xu, H. Saïdi, and R. Anderson. “Aurasium: Practical Policy Enforcement for Android Applications” The USENIX Security Symposium, pp.539-552, 2012.

[3] 신상호, 고국원, 심재환, 윤충은, "RF/LF 양방향 통신을 이용한 차량용 스마트 키 시스템 개발" 대한전기학회, pp.91-92, 2010.

[4] A. Bianchi, Y. Fratantonio, C. Kruegel, and G. Vigna “NJAS: Sandboxing Unmodified Applications in non-rooted Devices Running Stock Android”, The 5th Annual ACM CCS Workshop on Security and Privacy in Smartphones and Mobile Devices, pp.27-38, 2015.