

안드로이드 기반 스마트TV 셋톱박스의 효과적 파일 시스템 복구 방법에 관한 연구

김병준*, 한경식**, 손승일*,

한신대학교 정보통신학과*, 이노디지털(주)**

Reserach for Filesystem Recovery in Android based SMART TV Settop

Byung-Jun Kim*, Kyung-Sik Han**, Seung-II Sonh*,

Dept. of Information and Communication Hanshin University*, and INNODIGITAL Co., Ltd**

E-mail : jjun258@nate.com

요 약

모바일 시스템에 비해 안드로이드 기반 스마트TV 셋톱박스의 경우 직접 전원 연결을 통한 시스템 전원을 공급받기 때문에 갑작스러운 전원차단으로 인한 메모리 기반 파일시스템의 손상이 발생하게 된다. 이를 개선하기 위해 init의 중복 실행을 통해 스마트TV OS 프레임워크가 시작되기 전에 해당 파일 시스템의 손상 유무를 판단하여 시스템 체크를 수행하는 방법에 대해 연구한다.

ABSTRACT

Compare to mobile system, in case of Android SMART TV Settopbox, power cable connect directly to device, user's power shutdown makes file system damage on memory file system. This paper research the way of check file system before booting SMART TV OS framework.

키워드

안드로이드, 스마트TV, 파일 시스템 복구

I. 서 론

최근 스마트폰이 다양한 서비스와 어플리케이션을 제공하면서 IT 기기들의 스마트화를 이끌고 있다. TV산업 또한 스마트화의 흐름에 따라 네트워크에 접속하여 주문형 비디오(VOD)를 시청하거나 인터넷 기반의 쌍방향 서비스를 즐기기 위한 TV로 사용하기 위해 셋톱박스의 사용이 늘어나고 있다. 이에 따라 스마트 TV에 대한 다양한 서비스 개발 뿐 아니라 개발 및 문제점 개선에 대한 연구가 활발해 지고 있다[1].

셋톱박스는 모바일 시스템과 달리 직접 전원 연결을 통한 시스템 전원을 공급받기 때문에 갑

작스러운 전원차단으로 인한 플래시 메모리 기반 파일 시스템의 손상이 발생하게 된다. 이를 개선하기 위해 init의 중복 실행을 통해 스마트TV OS 프레임워크가 시작되기 전에 해당 파일 시스템의 손상 유무를 판단하여 시스템 체크를 수행하는 방법에 대해 연구한다.

II. 본 론

2.1 안드로이드 시스템의 동작

안드로이드가 부팅이 되면 먼저 Boot Loader를 실행하게 된다. Boot Loader는 메모리를 초기화하고 external RAM에 Boot Loader를 적재하며, file system, 네트워크 등을 설정하게 된다.

그 후 Linux의 부팅을 위해 커널을 로딩하고 이를 RAM에 적재하게 되면 리눅스 커널이 시작된다.

Linux Kernel은 해당 시스템이 동작하도록 인터럽트 컨트롤러를 초기화하고, 메모리 프로텍션, 캐시 그리고 스케줄링을 설정한다. 또한 커널은 Init 프로세스를 위해 루트 파일 시스템을 볼 수 있게 하고 초기 유저 스페이스 프로세스를 런칭하게 된다.

Init 프로세스는 시스템 서비스, 파일 시스템과 설정해야 할 파라미터들에 대한 내용을 가지고 있는 init.rc라는 파일을 파싱하고, 시스템 서비스 프로세스들을 시작하도록 한다.

Android Framework에서는 안드로이드 애플리케이션의 수명 주기를 관리하는 Activity Manager, 표준 widget을 담당하는 View System, 모든 응용프로그램과 관련된 화면을 담당하는 Window Manager, 시스템에서 동작 중인 응용프로그램의 정보를 담당하는 Package Manager 등의 안드로이드 서비스들을 시작하게 되고 안드로이드가 실행되게 된다[2].

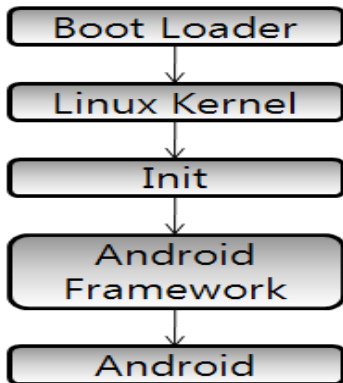


그림 1. 안드로이드 부팅 과정

본 연구에서는 Framework가 시작되기 전에 해당 파일 시스템의 손상 유무를 판단하여 시스템을 체크하고 복구하도록 하기 위해 init의 중복 실행 방식 사용하고자 한다.

2.2. 파일시스템 복구 방법의 제시 및 구조

Boot Loader와 Linux Kernel의 수행은 기본적인 안드로이드 수행 과정과 동일하지만 Linux Kernel이 수행된 후 init의 중복 실행 방식을 사용하게 된다.

첫 번째 init은 시스템 체크 및 복구를 수행하기 위한 init이다. 시스템 서비스 프로세스들이 구동되기 전이기 때문에 파일 시스템의 손상 유무를 판단하여 시스템을 체크하고 손상되었을 경우 복구를 수행할 수 있게 된다.

시스템 체크 및 복구가 완료되었다면 두 번째 init이 수행되어 일반적인 안드로이드에서의 init 역할인 시스템 서비스 프로세스들을 시작하게 되고, framework와 안드로이드가 실행된다.

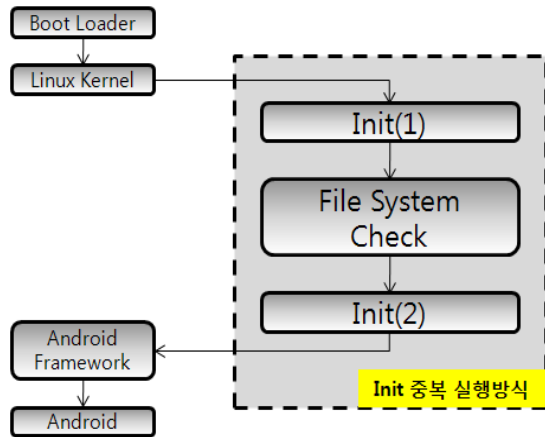


그림 2. init의 중복 실행 방식의 안드로이드 부팅 과정

2.3 제시방법의 활용방안

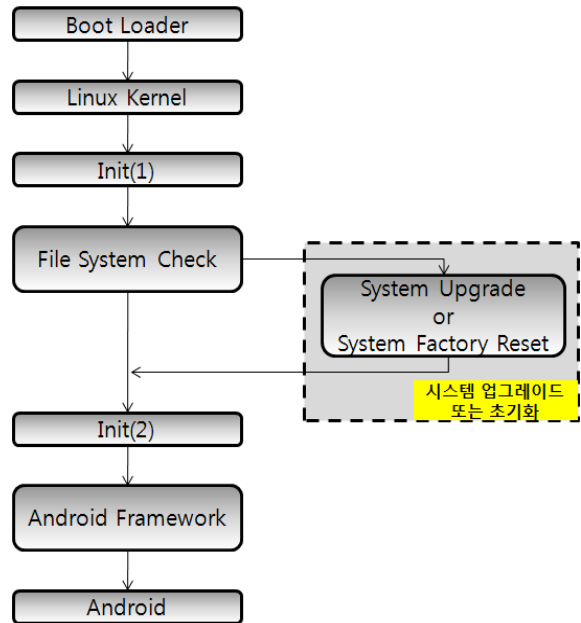


그림 3. 시스템 업그레이드 및 초기화 과정

일반적으로 안드로이드 시스템의 업그레이드는 시스템 서비스 프로세스가 실행되기 전에 이루어져야만 한다. 부팅을 하는 과정에서 시스템의 업그레이드나 초기화가 필요한지 미리 체크를 하더라도 이미 프로세스가 수행된 후 이기 때문에 OS의 재부팅 과정이 필요하게 된다 [2][3].

본 연구에서 제시한 init의 중복 실행 방식을 사용할 경우 첫 번째 init의 실행 과정에서 시스템 업그레이드 혹은 시스템의 초기화를 미리 체크를 하여 시스템 서비스 프로세스들이 실행되기 전에 진행할 수 있기 때문에 OS의 재부팅과 같은 번거로운 작업을 없앨 수 있게 된다.

III. 결 론

본 연구는 안드로이드 기반 스마트 TV 셋톱박스의 효과적 파일 시스템 복구 방법에 관한 연구로써 init의 중복 실행을 통해 스마트TV OS 프레임워크가 시작되기 전에 해당 파일 시스템의 손상 유무를 판단하여 시스템 체크를 수행한다. 이를 통해 추후 파일 시스템 업그레이드 및 시스템 초기화 부분과 연동하여 안정적인 파일 시스템을 구축하는데 활용될 수 있다.

참고문헌

- [1] 이환수, 조항정 “스마트 TV 기업의 플랫폼 경쟁전략” 한국정보사회학회 2012
- [2] 고현철, 유형목, 안드로이드의 모든 것 분석과 포팅, 한빛미디어, 2011
- [3] 남상엽, 강민구, 김인기, 안드로이드 분석과 실습, 상학당, 2009