RAW 이미지 뷰어 취약점 분석

황선홍, 이병천*
*중부대학교 정보보호학과

Vulnerability Analysis of RAW Image Viewer

Sunhong Hwang, Byoungcheon Lee*
*Department of Information Security, Joongbu University.

요 약

RAW 이미지 파일은 이미지 파일 포맷 중 하나로 디지털 카메라나 이미지 스캐너의 이미지 센서로부터 얻어진 데이터를 최소한만 처리한 원본 데이터를 포함하고 있다. RAW 이미지 파일은 일반적으로 널리 사용되는 손실 압축 이미지 파일 포맷인 JPEG, PNG 보다 높은 화질을 가지고 있지만 파일 크기가 크고 해당 파일을 PC에서 열어보기 위해서는 별도의 소프트웨어가 필수적으로 요구된다.

본 논문에서는 이러한 RAW 이미지 파일 처리를 위한 이미지 뷰어 소프트웨어들의 취약점을 효과적으로 분석하기 위해 RAW 이미지 파일의 구조를 분석하고, 어떤 필드의 값이 이미지 뷰어의 취약점을 도출하는데 유용한 값인지 조사하였으며 결론적으로 이를 통해 X사의 이미지 뷰어에서 임의 코드를 실행할 수 있는 취약점을 도출하는데 성공하였다.

I. 서론

RAW 이미지 파일[1]은 이미지 파일 포맷 중 하나로 DSLR(Digital Single-Lens Reflex camera), 디지털카메라, 이미지 스캐너 등의 이미지 센서로부터 얻은 데이터를 최소한만 처리한 원본 데이터를 포함하고 있다. RAW 이미지 파일들은 가공되지 않은 원본 상태이며 디지털 카메라 제조사에 따라 여러 가지 다른 확장자를 가지고 있다.

이러한 RAW 이미지 파일을 사용하는 데에는 많은 이점이 있는데, 가공되지 않은 상태의 파일이기 때문에 일반적인 이미지 파일보다 더 높은 화질을 갖고 있다. 예를 들어 JPEG(Joint Photographic coding Expert Group)와 비교하면 JPEG은 손실 압축 포맷인데 반해 RAW 포맷은 압축되지 않거나 무손실 압축을 사용하므로 이미지 원본을 그대로 가지고 있다. 또한 카메라의 RAW 파일들은 12bit 혹은 14bit의 명암 정보를 가지고 있으며, 가공된 TIFF나 JPEG 파일에 저장되어있는 감마 압축 8bit를 사용하지 않는다. 따라서 데이터는 그림자, 밝은 부분, 채도가 깊은 색에 더 정확성을 제공한다는 장점이 있다.

하지만 이러한 RAW 파일들은 일반적으로 JPEG 파일보다 약 2~6배 정도 용량이 크고, 카메라 제조업체 대부분이 RAW 포맷을 사용하고 있지만 표준 RAW 포맷이 별도로 지정된 것이 아니기 때문에 각자 다른 포맷을 사용하고 있다. 따라서 일반적인 PC에 설치된 이미지 뷰어를 이용하여 파일을 열어볼 수가 없기 때문에 별도의 소프트웨어 설치가 요구되는데, 본 논문에서는 이러한 RAW 파일을 열어보기 위해 널리 사용되는 이미지 뷰어들에 대한 취약점 분석을 위해 RAW 파일을 hex단위로 분석하여 각 필드의 정보를 조사하였고, 필드의 데이터를 고의적으로 조작하는 공격을 시도하였다. 결과적으로 X사의 이미지 뷰어에서 임의 코드를 실행할 수 있는 취약점을 발견하였다.

II. RAW 이미지 파일 포맷

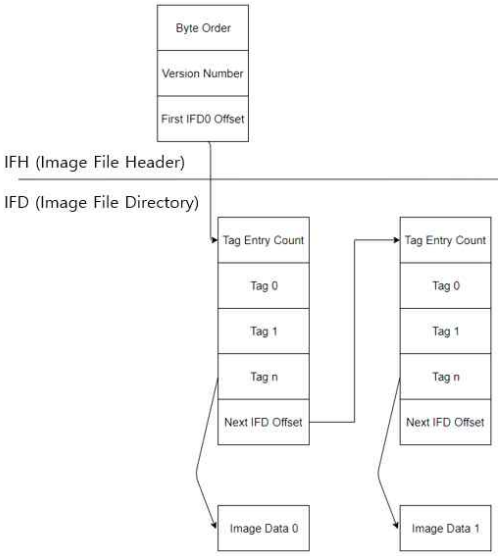
RAW 이미지 파일의 포맷은 [표 1]과 같이 DSLR 카메라 제조사별로 다르고, 정확한 포맷에 대한 정보는 카메라 제조업체의 지적재산권을 이유로 공식적으로 공개되지 않고 있다. 하지만 대부분의 RAW 이미지 파일이 TIFF(Tag Image File Format)를 기반으로 변형되었기 때문에 이를 토대로 분석이 가능하다.

[표 1] 제조사에 따른 RAW 이미지 파일 포맷

제조사	파일 확장자
FUJIFILM	.raf
Canon	.crw .cr2 .cr3
SAMSUNG	.srw
Nikon	.nef .nrw
Olympus	.orf
Adobe	.dng
PENTAX	.ptx .pef

2.1 TIFF 파일 분석

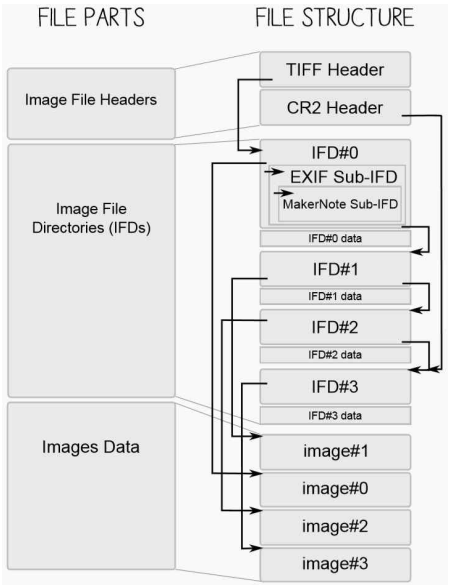
TIFF는 Aldus사와 마이크로소프트사가 공동으로 개발한 이미지 저장 포맷이며 사용자가 고쳐서 쓸 수 있는 유연한 특징이 있다. TIFF는 [그림 1]과 같이 IFH(Image File Header), IFD(Image File Directory), Bitmap Data등으로 구성된다.



[그림 1] TIFF 파일 구조

IFH는 2byte의 Byte-Order Identifier와 2byte의 TIFF 버전 정보, 4byte의 첫 번째 IFD에 관한 Offset정보를 가지고 있으며 IFD는 해당 IFD에 태그가 몇 개 존재하는지를 나타내는 값인 2byte의 NumDirEntries, 12byte의 태그 배열, 다음 IFD의 Offset정보를 나타내는 4byte의 값이 존재한다.

2.2 Canon사의 CR2 포맷 분석



[그림 2] CR2 포맷의 구조

CR2 포맷은 Canon사의 RAW 이미지 파일 포맷으로 Canon RAW Version 2를 의미하고, Canon사의 디지털 카메라에 의해 생성된 파일이다. CR2 포맷은 TIFF를 기반으로 하며 고품질, 비 압축, 비교적 크기가 큰 특징을 갖는다. CR2 포맷은 [그림 2]와 같이 크게 IFH, IFD, Image Data로 나뉘고, IFH에 TIFF Header 정보와 CR2 Header 정보가 위치한다. 또한 총 4개의 IFD를 가지고 있는 구조를 가지며 TIFF 구조에서 첫 번째 IFD인 IFD0에 Maker Note 등의 내용을 추가하였다. IFD0~2에 3개의 JPEG 이미지를 포함하고 있고, IFD3에 실제 RAW 이미지 파일이 위치한다.

III. RAW 이미지 뷰어 취약점 분석

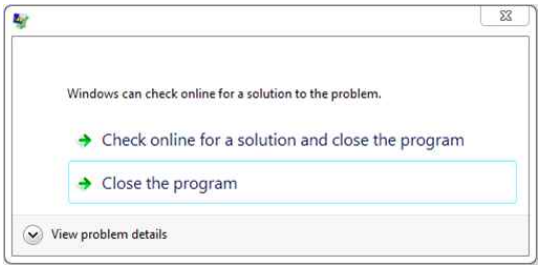
3.1 RAW 이미지 파일 변조 및 공격

앞서 조사한 내용을 바탕으로 CR2 파일을 이용하여 Smart 퍼징을 수행하였다. 취약점 분석의 대상이 되는 소프트웨어는 [표 2]와 같이 RAW 이미지 파일을 다루는 10개의 이미지 뷰어로 선정하였으며 본 논문에서는 CR2 포맷의 IFD0영역을 변조하며 Crash를 모니터링하였다.

[표 2] 취약점 분석 대상 이미지 뷰어

개발사	소프트웨어 명
Irfan Skiljan	IrfanView
FastStone Soft	FastStone
Sharpened Productions	File Viewer Plus
개인	Free Raw Viewer
SONY	RAW Viewer
개인	RawViewer
XnSoft	Xnview
GNU General Public License	UFRaw
Graphic-Region Development	Able RAWer
ideaMK	RAW Image Viewer

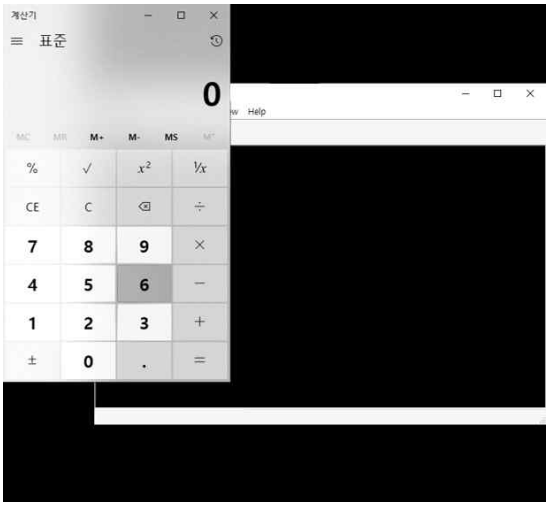
3.2 Crash 발생 및 원인



[그림 3] 이미지 뷰어에서 발생한 Crash

앞서 변조한 CR2 파일에 의해 10개의 이미지 뷰어 소프트웨어 중 5개의 소프트웨어에서 [그림 3]과 같이 운영체제의 Crash 메시지를 확인하였는데, Crash가 발생하는 전형적인 원인은 PC(Program Counter)레지스터가 부정확한 주소로 설정되거나 Buffer Overflow와 같은 버그 때문에 발생한다. 해당 Crash는 Model 태그의 Count 및 Tag_value를 정상적인 파일보다 큰 값으로 변조하였을 때 발생하였다. 해당 소프트웨어들에는 이에 대한 예외처리가 마련되어있지 않아 파일에서 정상적인 값보다 더 많은 값을 읽어오게 되었고 이는 Stack-Based Buffer Overflow를 야기하게 되었으며 X사의 소프트웨어에서는 이를 통해 PC레지스터를 조작하여 임의 코드를 실행할 수 있었다.

[그림 4]는 X사에서 발생한 Stack-Based Buffer Overflow 취약점을 이용하여 PC레지스터를 조작하는 것이 가능한 행위라는 것을 계산기를 실행함으로서 증명한 예시이다. 만약 공격자가 이를 악의적으로 조작한다면 X사의 소프트웨어가 설치되어 있는 모든 사용자를 대상으로 공격자가 원하는 모든 행위를 수행할 수 있다.

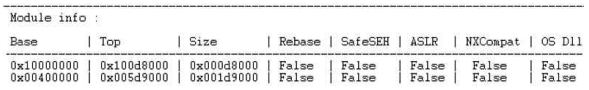


[그림 4] 이미지를 이용한 임의 코드 실행

3.3 취약점 개선 방안

Crash가 발생한 이미지 뷰어는 모두 CR2 포맷의 RAW 이미지 파일에서 변조된 Model 태그의 Count, Tag_value 값에 의해 발생하였는데 소프트웨어가 변조된 태그에 해당되는 값을 parsing할 때 보다 큰 값을 읽어오면서 발생하였다. 따라서 개발사는 이와 같이 비정상적인 값이 세팅되었을 때 parsing을 진행하지 않고 예외처리를 할 수 있는 루틴을 추가해야 될 것이다.

또한 일부의 이미지 뷰어에서 [그림 5]와 같이 메모리 보호기법을 확인하였을 때 DEP(Data Execution Prevention)와 같은 메모리 보호 기법이 적용되어있지 않아 쉘코드를 이용한 공격이 가능하였는데, 만약 DEP가 적용되어 있다면 쉘코드를 Stack 영역에 저장하더라도 실행권한이 없기 때문에 쉘코드가 실행되는 것을 방지할 수 있다. 따라서 소프트웨어의 메모리 보호기법의 적용 또한 요구된다.


Module info :

Base	Top	Size	Rebase	SafeSEH	ASLR	NXCompat	OS Dll
0x10000000	0x100d8000	0x000d8000	False	False	False	False	False
0x00400000	0x005d9000	0x001d9000	False	False	False	False	False

[그림 5] 메모리 보호기법 확인

IV. 결론 및 향후 연구

4.1 결론

RAW 이미지 파일의 포맷은 DSLR 카메라의 제조사에 따라 상이하며 제조사의 지적재산권을 침해한다는 이유로 일반인에게 공개되지 않고 있다. 하지만 대부분의 RAW 이미지 파일이 TIFF를 기반으로 변형된 포맷이기 때문에 이를 통해 분석이 가능하였고, 본 논문에서 조사한 CR2 포맷 RAW 이미지 파일의 IFD0의 Model 태그의 Count, Tag_value 값을 비정상적인 값으로 변조하여 10개의 이미지 뷰어 소프트웨어 중 5개의 소프트웨어에서 취약점 도출의 가능성을 확인하였으며 X사의 소프트웨어에서는 임의 코드를 실행이 가능하다는 것을 증명하였다. 이를 통해 개발사는 이미지의 비정상적인 필드 값에 대한 예외를 처리하는 루틴을 추가하고, 메모리 보호기법을 적용하여 사용자에게 좀 더 신뢰도 높은 소프트웨어를 제공하는 노력이 필요하다.

4.2 향후 연구

본 논문의 실험에서는 RAW 이미지 파일의 구조에 대해 조사하고, data model을 직접 세팅한 뒤 퍼징을 수행하여 취약점 분석을 진행하였다. 하지만 이러한 진행 방식은 파일 구조에 대해 직접 조사를 하는 과정에서 많은 시간을 필요로 하기 때문에 다양한 필드 및 파일을 조사하기에는 어려움이 있었다. 따라서 자동으로 필드의 값을 변경하며 코드 커버리지를 판단하여 취약점을 도출하는 방식의 퍼저를 개발할 계획이다. 이를 통해 취약점을 도출하는 과정에서 시간이 단축될 것으로 예상되며 이미지 파일 뿐만 아니라 다른 포맷의 파일을 이용한 소프트웨어 취약점 분석에도 유용하게 사용될 것으로 예상된다.

또한 본 논문의 실험 결과로 도출된 취약점은 KISA에 제보하여 보안 패치를 요구할 계획이다.

[참고문헌]

[1] 노광현. DSLR 카메라의 RAW 파일 포맷 분석. 한국컴퓨터정보학회 학술발표논문집, 2009, 16.2: 89-92.

[2] 김동진; 조성제. 멀티미디어 플레이어에 대한 퍼징기반 취약점 분석. 정보과학회논문지: 컴퓨팅의 실제 및 레터, 2011, 17.2: 98-107.

[3] OKAMOTO, Takeshi. SecondDEP: Resilient computing that prevents shellcode execution in cyber-attacks. Procedia Computer Science, 2015, 60: 691-699.

[4] GRIECO, Gustavo; CERESA, Martín; BUIRAS, Pablo. QuickFuzz: An automatic random fuzzer for common file formats. In: ACM SIGPLAN Notices. ACM, 2016. p. 13-20.