취약점 별 아티팩트 사례 분석을 통한 아티팩트 그룹핑 연구 : 어도비 플래시 플레이어 취약점을 이용하여

송 병 관*, 김 선 광**, 권 은 진***, 진 승 택****, 김 종 혁*****, 김 형 철******, 김 민 수*******

요 약

점차 고도화되는 사이버 공격에 의한 많은 침해사고로 피해가 증가하고 있다. 많은 기관 및 기업체에서는 사고 탐지를 위한 인프라만에 많은 자원을 투자하기에 초기대응에 미흡하다. 침해사고의 초기대응은 공격의 유입경로 파악이 우선이며, 이루어지고 있는 많은 사이버 공격은 소프트웨어 취약점을 대상으로 하고 있다. 따라서, 소프트웨어 취약점을 대상으로 윈도우 시스템의 아티팩트를 분석하고, 분석한 데이터를 분류하면 신속한 초기대응에 활용할 수 있다. 그러므로 소프트웨어 별 공격 유입 시 남는 아티팩트를 분류하여 침해사고 분석 시에 활용할 수 있는 아티팩트 그룹핑을 제시한다.

A Study on Artifact Grouping by Analyzing Artifact Case by Vulnerability : Using Adobe Flash Player Vulnerabilities

ByungKwan Song*, SeonKwang Kim**, EunJin Kwon***, SeungTaek Jin****

JongHyuk Kim*****, HyeongCheol Kim******, Minsu Kim*******

ABSTRACT

The damage is increasing due to many encroachment accidents caused by increasingly sophisticated cyber attacks. Many institutions and businesses lack early response to invest a lot of resources in the infrastructure for incident detection. The initial response of an intrusion is to identify the route of attack, and many cyber attacks are targeted at software vulnerabilities. Therefore, analyzing the artifacts of a Windows system against software vulnerabilities and classifying the analyzed data can be utilized for rapid initial response. Therefore, the remaining artifacts upon entry of attacks by software are classified, and artifact grouping is presented for use in analysis of encroachment accidents.

Key words : Artifact, Artifact Grouping, Incident Response, Adobe, Adobe Flash Player

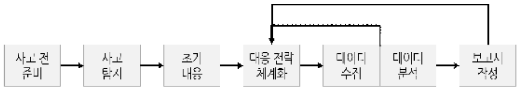
접수일(2018년 12월 7일), 게재확정일(2019년 3월 19일)

* 경기대학교 융합보안학과
** 중앙대학교 융합보안학과
*** 서울시립대학교 컴퓨터과학부
**** 호서전문학교 사이버해킹보안과
***** ㈜스틸리언
****** 삼정KPMG
******* 중부대학교 정보보호학과(교신저자)

1. 서 론

점차 고도화되는 사이버 공격에 의해 많은 침해사고가 일어나고 있으며, 그 피해 규모 또한 증가하고 있다. 이에 많은 기업과 기관에서는 침해사고에 대응하기 위한 프로세스를 구축하고 있다.

기업이나 기관에서는 침해사고 대응 프로세스를 위해 보안관제 팀과 침해사고 분석팀 등을 운영하고 있다. 규모는 다를 수 있지만, 대체로 일관된 절차를 가지고 있으며, KrCERT/CC(KOREA Computer Emergency Response Team Coordination Center)를 운영하는 한국인터넷진흥원에서는 (그림 1)과 같이 침해사고 대응을 7단계로 나누어 적용할 수 있도록 기업들에 안내하고 있다[1].



(그림 1) 침해사고 대응 7단계

하지만, 많은 기업들이 사고 전 준비와 사고 탐지를 위한 인력만을 구성하며, 그에 맞는 솔루션 제품만을 구매하고 이용한다. 이러한 시스템은 실제 사고가 발생할 경우, 탐지는 빠르게 이루어질 수 있으나, 탐지되지 않는다면 초기대응은 사고대응팀이 도착한 이후에나 이루어진다.

초기대응에서는 적절한 대응을 하기 위해 어디서부터 공격이 유입되었는지가 핵심이다. 실제로 사고대응팀이 도착하기 전 적절하지 못한 조치로 많은 증거를 훼손한다. 그리하여 사고대응팀이 도착한 이후에 공격의 유입 경로를 확보하는데 많은 시간을 소요하고 있다.

공격자는 다양한 방법으로 공격을 실행하고 있지만, 그중에서도 많이 사용하는 방법은 상용 소프트웨어의 취약점을 이용하는 방법으로, 대략 공격의 75%가 상용 소프트웨어의 취약점을 사용한 공격이기도 했으며[2], 많은 취약점 공격 도구에서도 상용 소프트웨어의 취약점을 노린 도구가 확인된다.

공격자가 주로 이용하는 소프트웨어의 종류는 <표 1>과 같으며, 해당 소프트웨어는 프로그램의 특정 기능 사용이나 버전에 따라서 정형화된 아티팩트(Artifact)를 생성한다. 이를 이용하여, 공격자가 사용하는 소프트웨어의 공격방법을 분석한 후, 정형화된 아티팩트를 소프트웨어나 취약점을 기준으로 그룹화하여 공격의 유입경로를 효율적으로 파악할 수 있다.

따라서, 본 논문에서는 소프트웨어 취약점을 이용한 침해사고에서 활용할 수 있는 아티팩트 그룹핑(Artifact Grouping)을 제시한다. 아티팩트 그룹핑은 어떤 한 소프트웨어를 기준으로 삼으며, 해당 소프트웨어의 취약점을 공격할 시 남는 아티팩트를 정형화하여 공격자의 공격 유입 경로를 효율적으로 판단하는데 목적을 둔다.

<표 1> 상용 소프트웨어의 종류

Type	Vendor	Product Name
Web Browser	Microsoft	Internet Explorer
		Edge
	Google	Chrome
Document	Microsoft	MS Office Word
		MS Office Excel
		MS Office PowerPoint
	Hancom	Hancom Office
	Adobe	Acrobat Reader
Player	Adobe	Flash Player

2. 관련연구

2.1 윈도우 아티팩트(Windows Artifact)

운영체제는 해당 시스템이 운영되면서 생기는 흔적을 남길 수 있게 설계되어있다. 사용자의 행위에 따라 생성되거나 시스템이 설계 단계에서부터 자동으로 남기게 되는 것들을 모두 아티팩트(Artifact)

라 부르며, 이것은 디지털포렌식 및 침해사고 분석에서 분석가의 분석을 뒷받침하는 역할을 한다.

아티팩트는 리눅스의 경우 로그형태로 남는 것이 대부분이다. 하지만 윈도우의 아티팩트는 다양한 형태로 남는 것이 특징이다. 또한, 아티팩트는 지속성으로 분류할 수 있는데, 전원이 있어야만 남는 휘발성 데이터와 시스템의 전원이 없이도 보조 기억 장치에 저장되는 비휘발성 데이터로 나눌 수 있다[3]. 본 논문에서는 침해사고 발생 이후의 분석에 초점을 맞추므로, 비휘발성 데이터만을 사용하며, 주요 분석 대상 아티팩트는 <표 2>와 같다.

<표 2> 아티팩트 종류

Type	Data
FileSystem artifact	파일 시스템 정보
Web artifact	웹 브라우저 사용 정보
Event log	윈도우 시스템 운영 정보
Prefetch	실행 파일 사용 흔적
Jump lists	최근 사용 문서, 자주 사용하는 문서
Temp Files	임시 사용 파일

2.1.1 파일시스템 아티팩트

파일 시스템 아티팩트는 디스크에 생성, 수정, 삭제되는 파일들을 기록한다. 어떤 파일 시스템을 채택하였는지에 따라 다르지만 대체로 아티팩트의 시간 흐름을 파악할 수 있는 주요 요소로 활용된다.

윈도우에서 가장 많이 사용하는 파일 시스템은 NTFS이다. NTFS 파일 시스템은 파일의 내부 정보를 가지고 있는 $MFT, 트랜잭션 로그를 기록하는 $LogFIle, 파일 변경상태를 기록하는 $UsnJrnl을 가지고 있으며, 이를 분석하면 파일의 생성, 수정, 삭제의 기본적인 시간 흐름을 파악할 수 있다[3].

2.1.2 웹 아티팩트

웹 아티팩트는 인터넷 익스플로러, 크롬, 엣지와 같은 웹 브라우저를 통해 생성되는 아티팩트로, 히스토리, 쿠키, 캐시, 다운로드 파일 등을 확인할 수 있다. 웹 아티팩트는 대개 웹 사이트 접속 시, 해당 사이트로부터 저장되는 데이터이다. 대다수의 공격은 웹을 통해 이루어지므로 침해사고에서 유용한 정보로 활용할 수 있다[3].

2.1.3 이벤트 로그

윈도우의 시스템 운용 중 특정 이벤트 발생시 시스템이 자체적으로 남기는 로그이다. 시스템의 주요 정보와 상태를 남기기 때문에, 시스템이 자체적으로 남긴 로그를 확인하기 위해선 필수적으로 파악해야 한다[4].

2.1.4 프리패치

윈도우의 프리패치는 실행파일이 실행될 때, 필요한 정보를 프리패치 파일에 저장한다. 윈도우 시스템이 부팅될 때, 모든 프리패치 파일을 로드하여 시스템 속도 향상에 기여하기 위해 고안되었다. 프리패치 분석을 통해 실행 횟수, 마지막 실행 시간, 참조 목록 등을 확인할 수 있다[3].

2.1.5 점프 목록

점프 목록은 윈도우 7 이후의 운영체제에서 최근 사용 문서 그리고 자주 사용하는 문서를 사용자의 편의에 맞추어 분류하여 제공한다. 점프 목록의 AutomaticDestination은 최근 사용한 문서 목록이며, CustomDestination은 자주 사용한 문서 목록이다. 점프 목록 분석을 통하여 열람한 문서가 어떤 파일인지, 언제 열람하였는지 등을 확인할 수 있다[3].

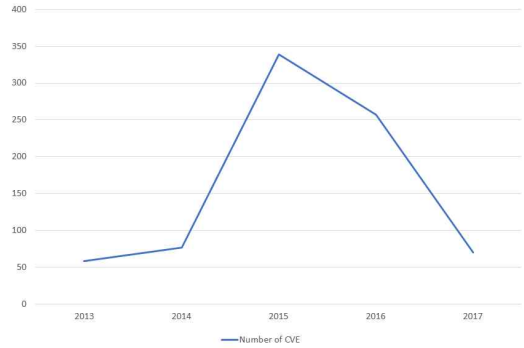
2.1.6 임시파일

임시파일은 응용프로그램이 특정 기능을 사용할 때 혹은 다른 응용프로그램과의 메모리 공유를 위해 사용한다. 임시파일은 그 특성상 쉽게 삭제된다는 단점이 있지만, 임시파일이 삭제되기 전에 신속하게 분석에 시도한다면 침해사고 분석 시에 필요한 많은 정보를 얻을 수 있다.

2.2 어도비 플래시 플레이어 취약점

어도비 플래시 플레이어는 다양한 소프트웨어 플랫폼에 내장되어 오디오, 비디오를 포함하여 역동적인 효과를 내는 소프트웨어이다. swf, flv 확장자를 가진 파일을 주로 로드하며, 높은 확장성으로 주로 웹 브라우저에서 사용하며, 문서형 소프트웨어에서도 활용될 수 있다.

어도비 플래시 플레이어는 높은 확장성으로 인해 공격자에게 다양한 공격방법을 제공하였고, 매년 수 많은 취약점을 쏟아내고 있다. 취약점 정보를 수집하여 공개하는 Mitre에서는 CVE Database를 제공하여 공개된 취약점 정보를 이용할 수 있도록 한다[5].



(그림 2) 어도비 플래시 플레이어의
연도별 CVE 번호 수

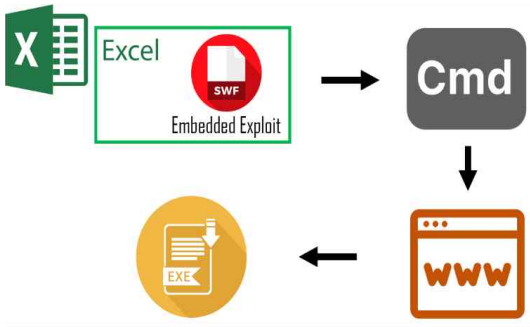
(그림 2)와 같이 어도비 플래시 플레이어의 취약점 추이를 보면, 2014년과 2015년에는 3배가량 증가한 것으로 보이며, 점차 그 보고된 취약점의 수는 줄어드는 것으로 보인다. 하지만, 많은 어도비 플래시 플레이어의 취약점은 낮은 버전으로 인해 야기되며, 실제 많은 사용자가 낮은 버전을 이용하기 때문에 그 위험성은 여전할 것으로 보인다.

따라서, 본 논문에서는 CVE 번호가 부여된 공개된 어도비 플래시 플레이어 취약점 3개를 분석하여 공격이 어떻게 진행되는지 살펴보고, 그에 따른 아티팩트 그룹핑을 하고자 한다.

2.2.1 CVE-2018-4878

CVE-2018-4878은 취약점 공격 도구인 Rig에서 제공하는 취약점이다[6]. UAF(Use-After-Free) 취약점을 이용한 공격으로 UAF 취약점은 소프트웨어가 메모리 사용을 위해 할당받았던 영역을 이용이 끝난 후 반환하게 되는데, 이때 이전에 반환했던 영역과 같은 크기를 할당받으려 요청하면, 이전에 사용했던 영역을 할당하게 된다[7]. 따라서, 이전에 사용했던 영역의 재사용으로 인해 원하지 않는 값을 참조한다. CVE-2018-4878은 공격 시나리오까지 공개되어 있기 때문에, (그림 3)과 같이 공개된 공격 시나리오를 이용하였다[8].

공격 시나리오는 가장 처음 MS 오피스의 엑셀에 어도비 플래시 플레이어를 사용하며 공격코드가 들어있는 swf파일을 내장시킨 후 피싱 메일이나 웹을 통해 엑셀 파일을 유포시킨다. 엑셀 파일을 다운로드한 사용자가 엑셀 파일을 실행시키면 어도비 플래시 플레이어가 swf파일을 자동으로 재생시키며, 재생되는 내용은 공격자가 심어놓은 공격코드이다. 본 논문에서 분석에 활용한 공격코드는 윈도우에 내장된 cmd.exe를 실행시켜 공격자가 미리 외부 유포를 위해 만들어 놓은 웹 페이지에서 파일을 다운로드한다.



(그림 3) CVE-2018-4878 공격 시나리오

2.2.2 CVE-2015-7645, CVE-2015-8651

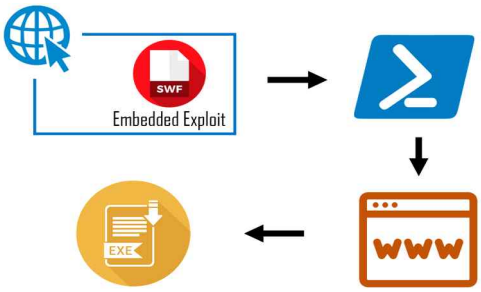
CVE-2015-7645, CVE-2015-8651은 모두 메모리 손상(Memory Corruption) 취약점으로, CVE-2015-7645는 버퍼 오버 플로우(Buffer over fl

ow)이며, CVE-2015-8651은 정수 오버 플로우(Integer Overflow) 취약점을 갖는다.

버퍼 오버 플로우는 프로그램이 실행될 때 할당된 메모리 영역에 값을 입력받는데 이때, 입력받는 값이 할당된 영역을 넘어서서 의도치 않은 값이 저장되거나 반환되는 취약점이다.

정수 오버 플로우는 정수형 변수의 값이 증가되어, 변수의 허용된 최댓값보다 큰 값이 되어 의도치 않은 값이 입력, 또는 반환되는 취약점이다.

CVE-2015-7645, CVE-2015-8651 모두 가장 기본적인 어도비 플래시 플레이어 취약점 공격 시나리오를 따르고 있다[9]. 공격 시나리오는 웹 페이지에 공격코드가 삽입된 swf파일을 실행시키는 html문서를 내장시킨다. 이후, 사용자가 해당 웹 페이지에 방문하게 되면, 내장된 html문서에 의해 어도비 플래시 플레이어가 swf파일을 자동으로 재생한다. swf파일에는 윈도우 파워 쉘(Windows Powershell)을 실행하여 공격자가 미리 구성한 웹 페이지에서 파일을 다운로드하여 실행시킨다.



(그림 4) CVE-2015-7645, CVE-2015-8651 공격 시나리오

3. 아티팩트 분석

3.1. 임시파일

운영체제는 표3과 같이 구성하여 진행했으며, 각 취약점 별 어도비 플래시 플레이어 버전과 swf를 불러오는 응용프로그램 구성은 별도로 하였다.

<표 3> 분석 대상 운영체제

O/S	Version
Windows 7	HomePremium SP1 x64
Windows 7	HomePremium SP1 x32
Windows 10	1507
Windows 10	1511

3.2 CVE-2018-4878

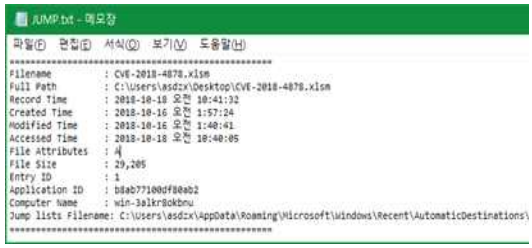
3.2.1 환경설정

CVE-2018-4878의 환경은 어도비 플래시 플레이어 28.0.0.137 버전을 사용하였고, 엑셀은 마이크로소프트 오피스 프로페셔널 2016 1809 버전을 사용하였다.

3.2.2 점프 목록

CVE-2018-4878은 엑셀에 내장된 swf 파일을 어도비 플래시 플레이어로 실행하는 공격 시나리오를 갖고 있다. 윈도우는 엑셀과 같은 문서를 실행하였을 때, 점프 목록을 남기게 되어있다.

(그림 5)를 보면 엑셀의 확장자인 .xlsm이 점프 목록에 들어가 있는 것이 확인된다. 점프 목록을 이용하여, 어떤 시간에 해당 문서를 열람하였는지 확인할 수 있으며, 또한 경로가 %Appdata% \Roaming\Microsoft\windows\Recent\AutomaticDestinations 이므로, 최근 실행한 파일임을 알 수 있다.



(그림 5) CVE-2018-4878 점프 목록

3.2.3 웹 아티팩트

(그림 6)에서는 윈도우에서 웹 아티팩트를 저장하는 Webcachev01.dat 파일을 이용한 웹 아티팩트를 분석한 그림으로 공격코드 실행 이후 임시인터넷 파일로 crossdomain.xml 파일이 남는다. 이 파일은 Adobe Flash Player나 Adobe Acrobat Reader에서 데이터 처리를 위해 특정 도메인에서 호스팅되는 컨텐츠를 요청하고, 액세스할 수 있게 하는 파일이다[10].

(그림 6) CVE-2018-4878 Web Cache

3.2.4 임시파일

어도비 플래시 플레이어는 캐시 파일을 저장하기 위해 NativeCache.directory를 사용한다. 어도비 플래시 플레이어가 사용되면 NativeCache. directory가 수정된다. 또한, Setting.Sol은 설정 정보를 저장한다[10].

(그림 7) CVE-2018-4878 NativeCache

<표 4> 임시 파일

Artifact	Location
Flash Cache	%Appdata%Roaming\Adobe\Flash Player\NativeCache
Shared Objects	%Appdata%Roaming%Macromedia\Flash Player \#Shared Objects
Setting Info.	%Appdata%Roaming\Macromedia\Flash Player \macromedia.com\support\flashplayer\sys

3.3 CVE-2015-7645, CVE-2015-8651

3.3.1 환경설정

CVE-2015-7645의 환경은 어도비 플래시 플레이어 18.0.0.241 버전을 사용하였고, 웹 브라우저는 마이크로소프트 인터넷 익스플로러 11버전을 사용하였으며, CVE-2015-8651의 환경은 어도비 플래시 플레이어 18.0.0.268 버전을 사용하였고, 웹 브라우저는 마이크로소프트 인터넷 익스플로러 11버전을 사용하였다.

3.3.2 웹 아티팩트

CVE-2015-7645와 CVE-2015-8651을 비교 분석해 보면, 공통적으로 Crossdomain.xml과, swf 파일, html문서가 남는다. Crossdomain.xml이 남은 것을 통해, 사용한 어도비 플래시 플레이어가 데이터 처리를 위해 특정 도메인과 통신을 한 것을 알 수 있다. 또한, 브라우저를 이용한 swf파일 로드 시에는 임시 인터넷 파일에 swf파일이 남는 것을 확인할 수 있다.

(그림 8) CVE-2016-7645 xml file

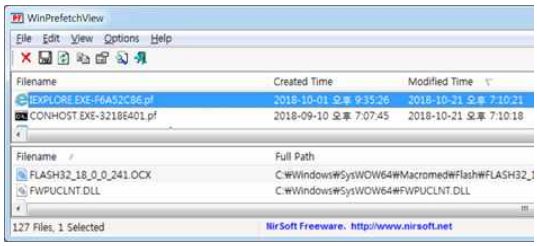
(그림 9) CVE-2015-8651 xml file

(그림 10) CVE-2015-7645 swf file

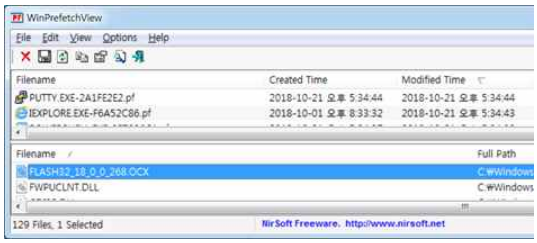
(그림 11) CVE-2015-8651 swf file

3.3.3 프리패치

프리패치 정보를 확인해보면, IEXPLORER.EXE의 참조목록에 설치되어있는 어도비 플래시 플레이어의 버전에 맞춰 FLASH32_18_0_0_241.OCX, FLASH32_18_0_0_268.OCX 파일이 있는 것을 확인할 수 있다. 인터넷 익스플로러에서 어도비 플래시 플레이어를 활용한 것을 알 수 있으며, 어도비 플래시 플레이어의 버전 정보도 확인할 수 있다.



(그림 12) CVE-2015-7645 프리패치



(그림 13) CVE-2015-8651 프리패치

3.4 아티팩트 그룹핑

아티팩트 그룹핑은 해당 소프트웨어의 동작으로 인해 특정하게 남아야 한다. 본 논문에서 분석한 어도비 플래시 플레이어 취약점을 가진 세 개의 CVE 모두가 남기는 아티팩트는 crossdomain.xml이다. 또한, 어도비 플래시 플레이어가 실행될 때마다 남기는 캐시 파일, 공유 객체 등도 공통적인 아티팩트이다.

반면, 같은 어도비 플래시 플레이어의 취약점이어도 매개하는 소프트웨어에 따라 다른 아티팩트가 남는다. 문서를 매개로 로드하는 플래시 파일은 점프리스트를 남기며, 웹 브라우저를 통해 로드하는 플래시 파일은 프리패치의 참조리스트에 들어간다.

따라서, 어도비 플래시 플레이어의 아티팩트 그룹핑 방법은 (표 5)와 같이 두 가지로 나눌 수 있다. 웹 브라우저를 이용한 플래시 파일과 문서 편집 소프트웨어를 이용한 플래시 파일은 웹 캐시와 설정 정보 등이 동일하게 남는다. 이러한 아티팩트를 분석하여 어도비 플래시 플레이어 취약점 이용이라는 추정이 가능하며, 문서 편집 소프트웨어와 웹 브라우저를 이용한 아티팩트가 각각 다르므로, 어떤 소프트웨어를 매개로 플래시를 실행하였는지에 대한 결론도 도출이 가능하다.

<표 5> 아티팩트 그룹핑

Type	Document	Web Browser
Artifact	Jumplist	Prefetch
	Setting Info.	Setting Info.
	document file	html document
	Web cache	Web cache

4. 결 론

대부분의 상용 소프트웨어는 버전이 업데이트 되어도 동작 방식이 완전히 변하지는 않는다. 따라서, 프로그램이 실행되는 동안 발생하는 아티팩트도 크게 다르지 않으며, 소프트웨어 취약점을 이용한 공격으로 남는 아티팩트도 일관성이 있다. 개별적인 아티팩트를 분석하는 것이 아닌 여러 아티팩트들의 연계성에 초점을 맞추어 소프트웨어별 아티팩트 그룹핑을 하는 것만으로도 공격의 유입 경로를 파악하기에 효율적이다. 아티팩트 그룹핑을 이용한 효과적인 공격 유입 경로 파악으로 침해사고 분석 시, 신속한 대처를 야기할 수 있다.

본 논문에서는 어도비 플래시 플레이어의 3가지 CVE 번호에 대해서만 분석하였지만, 향후 인터넷 익스플로러, 크롬, 엣지, MS 오피스, 한글 등 다양한 상용 소프트웨어의 취약점과 아티팩트 분석을 통하여 그룹핑을 한다면 침해사고 분석 시 유용하게 활용할 수 있을 것이다.

참고문헌

[1] 방송통신위원회, 한국인터넷진흥원, '침해사고 분석절차 안내서', 한국인터넷진흥원, pp. 12-14, 2010.

[2] Gartner, "Now is the time for security at application level", http://www.gartner.com/id=487227, 2005.

[3] 전상준. "윈도우즈 시스템 포렌식." 정보보호학회지, 26.5, pp. 6-16, 2016.

[4] Microsoft, "About Event Logging" GitHub. https://msdn.microsoft.com/ko-kr/library/win dows/desktop/aa363632(v=vs.85).aspx, 2018.

[5] Mitre, "Common Vulnerabilities and Exposures" "https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=Adobe+flash", 2018.

[6] Rod Soto & Kevin Stear, "Rig Exploit kit delivering GandCrab Ransomware via Adobe CVE-2018-4878", JASK LABS, 2018.

[7] J. Caballero, G. Grieco et al, "Undangle:Early Detection of Dangling Pointers in Use-after-free and Double-free Vulnerabilities," ISSTA, 2012

[8]Warren Mercer & Paul Rascagneres, "Flash 0-Day In The Wild:Group 123 At the Controls", Cisco, https://blog.talosintelligence.com/2018/02/group-123-goes-wild.html, 2018.

[9] Ana Dascalescu, "How Flash Vulnerabilities Expose You To Attacks", https://heimdalsecurity.com/blog/adobe-flash-vulnerabilities-security-risks/, Heimdal Security, 2017

[10] Apurva Udaykumar, "Setting A Crossdomain.xml File For Http Streaming", https://www.adobe.com/devnet/adobe-media-server/articles/cross-domain-xml-for-streaming.html, 2012.

[11] Adobe, "Removing Adobe Flash Access data files", https://helpx.adobe.com/x-productkb/multi/removing-flash-access-data-files.html, 2016.

〔저자소개〕

송 병 관 (ByungKwan Song)

현 재 경기대학교 융합보안학과
email : qudrhks333@gmail.com

김 선 광 (SeonKwang Kim)

2017년 수원대학교 컴퓨터공학 학사
현 재 중앙대학교 융합보안학과 석사과정
email : blueksg0307@gmail.com

권 은 진 (EunJin Kwon)

현 재 서울시립대학교 컴퓨터과학부
email : gej48443@gmail.com

진 승 택 (SeungTaek Jin)

2019년 호서전문학교 사이버해킹보안과 학사
email : jinst089@gmail.com

김 종 혁 (JongHyuk Kim)

현 재 ㈜스틸리언 연구원
email : jhkim@stealien.com

김 형 철 (HyeongCheol Kim)

2019년 홍익대학교 컴퓨터공학 학사
현 재 삼정KPMG IRM
email : hyeongcheolkim@kr.kpmg.com

김 민 수 (Minsu Kim)

2004년 컴퓨터공학사
2012년 경호안전학석사
2015년 산업보안학박사
현 재 중부대학교 정보보호학과 겸임교수

email : fortcom@hanmail.net