정 보 보 호 학 회 지
제34권 제5호, 2024. 10

Windows 환경에서의 Phobos 랜섬웨어 특징 및 동작 분석 연구

안 원 석*, 박 명 서**

요 약

랜섬웨어로 인한 피해가 전 세계적으로 점차 늘고 있다. 시간이 지나며 랜섬웨어도 점차 발전하며 암호화 방식이나 수익 구조의 특징으로 인한 랜섬웨어의 특징이 점점 다양해지고 있다. 본 논문에서는 2024년도에도 활발히 활동하고 있는 Phobos 랜섬웨어를 대상으로 랜섬웨어 동향 관점에서의 Phobos 랜섬웨어의 특징을 살펴보고 랜섬웨어의 동작 및 암호화 파일을 분석하였다. 또한, 랜섬웨어의 특징을 바탕으로 암호화 파일의 복호화 가능 여부를 판단하였다.

Ⅰ. 서 론

랜섬웨어의 피해 사례가 전 세계적으로 늘고 있다. 2024년도 2분기 랜섬웨어 공격이 1분기에 비해 18% 증가하였으며, 국내의 경우 10배 증가하였다[1]. 이처럼 증가하고 있는 랜섬웨어의 피해를 최소화하기 위해 랜섬웨어를 대상으로 한 분석 연구가 필요하다.

본 논문에서는 Windows 환경에서의 Phobos 랜섬웨어를 대상으로 분석 연구를 진행한다. Phobos 랜섬웨어는 2019년도에 처음 발견된 이후 2024년 현재에도 활발히 활동하고 있는 랜섬웨어 중 하나이다. 최근 피해 사례로 2024년 7월 나이지리아의 클라우드 서비스 회사를 대상으로 한 공격 사례가 있으며[2], Faust, Eight, Elbie, Devos와 같은 다양한 변종이 존재한다. 그리고 2024년도에 활발히 활동 중인 랜섬웨어 그룹 중 하나인 “8base” 그룹이 Phobos와 연결되어 있다고 알려지는[3] 등 위험성이 높은 랜섬웨어이다.

본 논문에서는 해당 랜섬웨어의 특징을 랜섬웨어의 동향 관점에서 살펴보고 암호화 전 동작, 암호화 동작 및 암호화 파일의 특징에 대해 설명한다. 그리고 Phobos 랜섬웨어의 특징을 통해 원본 파일의 복호화 가능 여부를 판단한다.

Ⅱ. 관련 연구

랜섬웨어에 대한 사람들의 관심이 높아지며 랜섬웨어 대상 분석 연구 또한 많이 진행되었다. 차해성 외 2인은 Windows 환경에서 Bianlian 랜섬웨어를 대상으로 역공학 분석을 통해 작동 원리를 분석하고 복호화 방안을 제시하였다[4]. 강수진 외 5인은 Ragnar Locker 랜섬웨어의 동작 및 암호화 과정을 분석하고 키 재사용 공격을 통해 복호화하는 방안을 제시하였다[5]. 이영주는 스트림 암호 ChaCha, Salsa20 2종의 동작 과정을 분석하고 스트림 암호 기반 랜섬웨어인 Conti, Ragnar, BlackMatter, Babuk Locker를 대상으로 동작 과정 및 암호 기능을 분석하였다[6]. Luis 외 1인은 Salsa20 기반 랜섬웨어에 대하여 메모리에서 키 추출 및 파일당 하나의 키 랜섬웨어 암호화 키 복구를 위한 방법을 연구하고 Sodinokibi 랜섬웨어 대상으로 실험을 진행하였다[7]. Karan 외 3인은 Clop 랜섬웨어에 대하여 정적, 동적으로 분석하여 랜섬웨어의 동작 방식을 분석하였다[8] Fabrizio 외 1인은 Jigsaw, CryptoLocker, Petrwrap, TeslaCrypt 총 4종의 랜섬웨어를 대상으로 암호화 모델, 암호화 알고리즘, 키 생성 방법을 분석하였다[9].

* 한성대학교 IT융합공학부 (학부생, aws0918@hansung.ac.kr)
** 한성대학교 융합보안학과 (조교수, pms91@hansung.ac.kr)

Ⅲ. 랜섬웨어의 동향 관점에서의 Phobos 랜섬웨어의 특징

최근 많은 랜섬웨어 그룹이 사용하는 수익 모델은 제작자에게 일정 비용을 지불하면 랜섬웨어를 서비스 형식으로 제공하는 RaaS (Ransomeware as a Service) 방식의 수익 구조로[10], Phobos 랜섬웨어의 수익 구조 역시 RaaS 방식이며 서비스 방식으로 제공한다는 특성상 같은 Phobos 랜섬웨어라도 암호화 파일 확장자 또는 랜섬 노트의 구조 변화가 존재한다.

최근 랜섬웨어의 공격 방식은 단순히 파일을 암호화하고 금전을 요구하는 방식에서 파일을 암호화하고 추가적인 공격 행위를 가하는 다중 갈취 공격을 사용한다. Phobos 랜섬웨어의 경우 파일 암호화와 더불어 원본 파일을 빼돌려 파일 복호화와 데이터를 유출하지 않는 조건으로 금전을 요구하는 이중 갈취 공격을 사용한다.

BlackBasta, PLAY, BlackCat 등 다양한 랜섬웨어에서 파일 전체가 아닌 일부만 암호화하는 간헐적 암호화 방식을 사용한다. 이는 암호화에 걸리는 시간을 줄이며 동시에 파일을 온전히 사용할 수 없게 하며 평문이 남아있기 때문에 보안 솔루션이 탐지하기 어려운, 랜섬웨어의 관점에서 효율적인 암호화 방안이다. Phobos 랜섬웨어의 경우 파일 크기에 따라 파일 전체를 암호화하거나 앞선 간헐적 암호화 방식을 사용하여 파일을 암호화한다.

Ⅳ. Phobos 랜섬웨어의 동작 분석

Phobos 랜섬웨어의 동작 분석을 위해 [표 1]과 같이 환경을 구성하였다. Windows 10 환경에서 분석을 진행하였고, 파일의 데이터를 확인하기 위해 HxD Editor를 사용하였고 랜섬웨어 작동 분석을 위해 Process Monitor를 사용하였다. Phobos의 전반적인 동작 과정 분석을 위해 동적 분석을 수행하였으며, 이를 위해 활용한 디버거는 x64dbg이다. 또한, Phobos 랜섬웨어에 의해 파기된 파일을 디스크 상에서 복원 가능한지 확인하기 위해 FTK Imager를 이용한 디스크 이미징을 수행하였다.

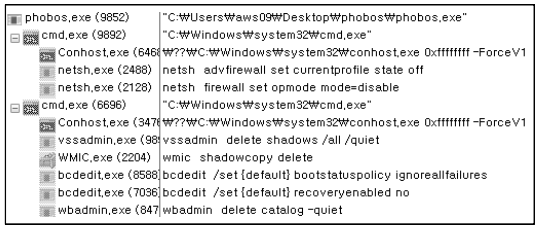
〔표 1〕 랜섬웨어 분석 환경

Category	Name and Version
OS	Windows 10
Ransomeware Sample	SHA-256: 45de59851d68929632346d6f894dc8c1b6a5c4197db83c2e33c60631efc0b39f
Viewer	HxD Editor v.2.5.0.0
	Process Monitor Published June 20, 2024
Debugger	x64dbg snapshot_2024-06-03_21-20
Disk Imaging	FTK Imager v.4.7

4.1. 암호화전 동작 분석

암호화 실행 전 Phobos 랜섬웨어는 효과적인 공격을 위한 환경을 구성한다. 먼저 자기 자신을 시작 프로그램으로 등록하여 재부팅 했을 때 랜섬웨어가 작동하도록 한다.

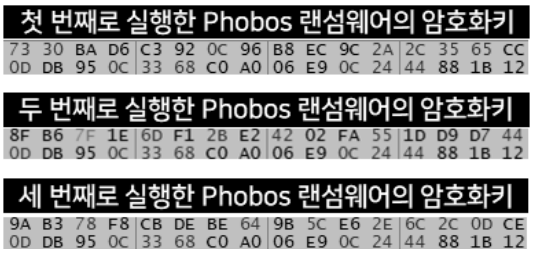
이후 명령 프롬프트를 이용하여 환경을 구성한다. “netsh advfirewall set currentprofile state off” 명령어와 “netsh firewall set opmod mod=disable” 명령어를 이용하여 방화벽을 종료하며 “vssadmin delete shadows /all /quiet” 명령어와 “wmic shadowcopy delete”, “bcdedit /set {default} bootstatuspolicy ignoreallfailures”, “bcdedit /set {default} recoveryenabled no”, “wbadmin delete catalog –quiet“ 명령어를 사용하여 Windows 백업을 삭제한다. 이는 Process Monitor의 Process Tree 기능을 사용하여 [그림 1]과 같이 확인할 수 있다.



〔그림 1〕 Process Tree로 확인한 Phobos 랜섬웨어의 명령어 사용

4.2. 암호화 동작 분석

Phobos 랜섬웨어의 암호화에는 AES-256-CBC를 사용하며 평문이 16바이트 블록 단위가 되도록 0x00으로 남는 채워 넣는 Zero-byte Padding을 사용한다. 32 바이트의 암호화키의 하위 16바이트는 “0x0ddb950c3368c0a006e90c2444881b12” 라는 고정된 값이며



〔그림 2〕 랜섬웨어 실행 시 암호화키의 변화

상위 16바이트는 [그림 2]와 같이 랜섬웨어가 실행될 때마다 변동하였다. IV 값은 암호화할 때마다 변화하였다.

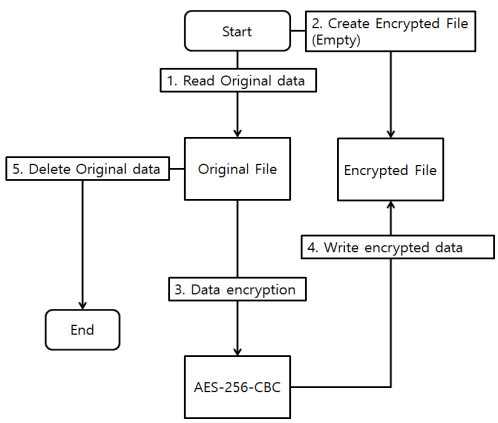
한 번 설정된 32바이트 암호화키는 랜섬웨어가 작동하는 동안은 모든 파일에 대하여 같은 암호화키를 사용해 암호화하였다. 그리고 암호화 파일의 크기에 따라 파일 전체를 암호화하거나 일부만 암호화하는 간헐적 암호화 방식을 사용하며, 재시작 시 암호가 손상될 수 있다는 랜섬 노트의 언급과 달리 암호화에 사용된 암호화키로 문제없이 복호화 할 수 있었다.

암호화키는 암호화가 진행되는 동안 메모리에서 확인할 수 있었으나, 암호화가 마무리되고 랜섬 노트가 발생한 이후에는 메모리에서 확인할 수 없었다.

Ⅴ. 암호화 파일 분석

5.1. 암호화 파일 특징

Phobos 랜섬웨어는 암호화 데이터를 원본 파일에 그대로 덮어 쓰지 않는다. 암호화 파일 생성은 [그림 3]과 같이 먼저 원본 파일을 열어 데이터를 읽고, 암호화한 데이터를 저장할 새로운 파일을 생성한다. 새로운 파일명은 원본 파일명에 “.id[C2C4E29C-3483].[recovery8files@onionmail.org].8base” 라는 문자를 추가한 형태가 된다. 이 때 새로운 파일명은 Phobos 랜섬웨어마다 [표 2]와 같이 차이가 존재하였다. 생성된 파일에 암호화 데이터를 쓴 후 원본 파일을 삭제하는 것으로 암호화를 마무리한다.

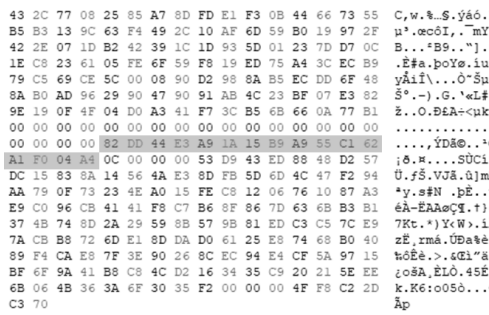


〔그림 3〕 암호화 파일 생성 프로세스

〔표 2〕 랜섬웨어 샘플 별 파일 이름

Sample SHA-256 Hash	File Name
45de59851d68929632346d6f894dc8c1b6a5c4197db83c2e33c60631efc0b39f	.id[C2C4E29C-3483].[recovery8files@onionmail.org].8base
4ff314143f6fea359946a81034ec04a4f515998fc23c6937bc5d032b02f01bea	.id[C2C4E29C-3368].[thekeyishere@cock.li].Elbie
2a8353551d099c78ac100b44718a691142f8cc7879b47e842ee8491426e15c08	.id[C2C4E29C-2822].[frankmoffit@aol.com].eight
00723db8c6513a9b8a79b8b8cc7d9da9f23a8a5454149ed12768937ca15d1a47	.id[C2C4E29C-3546].[getdataback@rambler.ru].faust

암호화 파일 내부는 [그림 4]와 같다. 암호화된 데이터 뒤로 20바이트 길이의 0x00 데이터가 저장되어 있고 그 뒤로 청색 표시된 내용과 같이 암호화에 사용된 16바이트 길이의 IV가 저장되어있다. 그 뒤로는 암호화된 데이터에 대한 패딩 길이가 4바이트 길이로 저



〔그림 4〕 암호화된 데이터

장되며, 그 뒤로 RSA-1024 알고리즘으로 암호화 된 AES 키가 저장된다.

5.2. 간헐적 암호화

3장에서 언급한 것과 같이 Phobos 랜섬웨어는 파일의 크기에 따라 암호화하는 방식이 다르다. 실험 결과 1.5 MB 크기를 기준으로 디스크 할당 크기가 1.5 MB 미만 파일 크기를 가지는 데이터는 3장 1절과 같이 파일 전체가 암호화되며 디스크 할당 크기가 1.5 MB 이상 파일은 원본 데이터의 일부분만 암호화하는 간헐적 암호화 방식을 사용한다.

Phobos 랜섬웨어에서의 간헐적 암호화 방식은 다음과 같다. 암호화되는 원본 데이터 블록의 크기는 256 KiB이며 원본 파일을 3등분하여 처음과 끝 그리고 중간 위치의 데이터 총 세 블록이 암호화된다. 다시 말해 총 256*3 KiB 크기의 데이터가 암호화된다. 암호화된 데이터들은 파일의 끝에 [그림 4]과 같은 형식으로 저장되며 암호화된 부분에 해당하는 원본 데이터는 [그림 5]과 같이 0x00으로 채워진다.

암호화된 파일 데이터와 원본 파일 데이터를 비교하였을 때 동일한 Offset에 동일한 평문이 존재하였다. 따라서 암호화된 부분을 복호화하고 이를 3등분 하여 0x00으로 채워진 부분에 각각 채워주면 온전한 원본 파일이 완성된다.

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0003FF60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0003FF70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0003FF80	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0003FF90	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0003FFA0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0003FFB0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0003FFC0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0003FFD0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0003FFE0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0003FFF0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00040000	55	8B	EC	6A	FF	68	61	0B	58	00	64	A1	00	00	00	00
00040010	50	83	EC	3C	A1	14	4E	60	00	33	C5	50	8D	45	F4	64
00040020	A3	00	00	00	00	89	4D	CC	8B	45	08	89	45	EC	8B	4D
00040030	EC	8B	51	10	52	8B	45	EC	8B	48	0C	51	8D	55	E8	52
00040040	8B	4D	CC	81	C1	74	01	00	00	E8	02	CC	0D	00	C7	45
00040050	FC	00	00	00	00	8B	45	EC	8B	48	0C	8D	14	CD	48	5F
00040060	5A	00	52	8B	45	CC	8B	88	70	01	00	00	E8	AF	79	02
00040070	00	50	8B	4D	CC	8B	89	70	01	00	00	E8	40	12	02	00
00040080	89	45	F0	8D	4D	E8	E8	E5	40	FC	FF	8B	55	F0	D1	E2
00040090	3B	C2	7F	2F	8D	4D	E8	E8	D4	40	FC	FF	85	C0	74	23
000400A0	51	8B	CC	89	65	D4	8D	45	E8	50	E8	21	40	FC	FF	89
000400B0	45	C8	E8	79	E3	01	00	83	C4	04	89	45	C4	83	7D	C4
000400C0	00	75	1D	8B	4D	0C	C7	01	FF	FF	FF	FF	C7	45	FC	FF
000400D0	FF	FF	FF	8D	4D	E8	E8	15	40	FC	FF	E9	03	01	00	00
000400E0	8D	4D	E8	E8	D8	3D	FC	FF	83	7D	F0	08	75	5E	51	8B
000400F0	CC	89	65	D0	8D	55	E8	52	E8	D3	3F	FC	FF	89	45	C0
00040100	E8	4B	E5	01	00	83	C4	04	89	45	B8	89	55	BC	8B	45
00040110	B8	89	45	DC	8B	4D	BC	89	4D	E0	8B	55	EC	8B	42	0C

〔그림 5〕 간헐적 암호화된 데이터 예시

해당 데이터를 복호화하면 원본 데이터와 원본 데이터의 시작 Offset이 저장된다 [그림 6]의 녹색 표시된 데이터가 시작 Offset, 청색 표시된 데이터가 원본 데이터로 데이터 시작 오프셋에서 256 KiB 크기, 즉 0x40000 크기에 해당하는 데이터를 채워 넣어 주면 복호화가 가능하다.

암호화가 마무리되면 mshta.exe를 통해 info.hta 라는 이름의 랜섬 노트를 출력하여 감염 사실을 알리며 금전을 요구하는 것으로 랜섬웨어의 동작이 마무리된다.

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00000000	00	00	00	00	01	00	00	00	0F	BC	77	AF	03	00	00	00
00000010	00	00	04	00	5F	34	7F	8A	38	00	0C	00	57	7F	38	6C
00000020	00	00	00	00	00	00	00	00	AA	42	0E	00	00	00	00	00
00000030	00	C8	26	00	00	00	00	00	4D	5A	90	00	03	00	00	00
00000040	04	00	00	00	FF	FF	00	00	B8	00	00	00	00	00	00	00
00000050	40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000070	00	00	00	00	F8	00	00	00	0E	1F	BA	0E	00	B4	09	CD
00000080	21	B8	01	4C	CD	21	54	68	69	73	20	70	72	6F	67	72
00000090	61	6D	20	63	61	6E	6E	6F	74	20	62	65	20	72	75	6E
000000A0	20	69	6E	20	44	4F	53	20	6D	6F	64	65	2E	0D	0D	0A

〔그림 6〕 복호화된 간헐적 암호화 데이터 예시

5.3. 복호화 가능 여부

암호화키의 경우 4장 2절에서 설명한 것처럼 랜섬노트 발생 이후 메모리에서 식별 불가능하다는 한계가 존재한다. 또한, 32바이트 키 중 상위 16바이트가 랜섬웨어 동작마다 변화하기 때문에 키를 재사용하여 복호화할 수 없다. 하위 16바이트가 고정된 값이므로 32바이트 전체를 전수조사할 필요는 없지만, 16바이트 길이를 전수조사하는 것 역시 사실상 불가능하다. 다행히 PC를 재부팅 하면 암호 키가 손상될 수 있다는 랜섬 노트의 언급과는 다르게 재부팅 이후에도 문제없이 복호화가 되었으나 앞선 이유로 Phobos 랜섬웨어를 직접 복호화하는 것은 사실상 힘들 것으로 보인다.

5장의 설명대로 Phobos 랜섬웨어는 원본 파일에 암호화 데이터를 덮어쓰지 않고 새로운 파일을 생성하고 기존 파일을 삭제한다는 특징이 있다. 직접 데이터를 복호화하지 않고 디스크 이미지 도구로 삭제된 원본 파일을 복구하거나 디스크 이미지 내에서 원본 파일의 데이터를 탐색하여 원본 파일을 복원하는 방법을 시도하였다.

FTK Imager를 사용하여 삭제된 파일을 복원할 수 있는지 시도하였으나 [그림 7]과 같이 시간이 지나면 복원 가능한 데이터가 사라진 것을 확인할 수 있다. 디스크 전체를 이미징하여 원본 파일의 데이터가 존재하

〔그림 7〕 암호화 진행에 따른 복구 가능한 데이터 식별 여부

는지 탐색하였으나 이 역시 랜섬웨어가 암호화에서 제외한 시스템 파일과 간헐적 암호화로 남아있는 일부 원본 데이터를 제외하고는 원본 파일의 데이터는 확인 불가능하다.

암호화된 AES 키를 복구하는 것 역시 시도하였으나 이 역시 암호화 이후 메모리에서 식별 불가능하여 AES 키를 복호화하는 방안 역시 사용할 수 없었다.

결론적으로 암호화 이후 메모리와 디스크에서 AES 암호화 키 RSA 암호화 키 그리고 삭제된 원본 데이터 전부 확인할 수 없어 복호화가 사실상 불가능하다고 판단된다.

VI. 결 론

이전에는 단순히 파일을 암호화만 하여 사람들에게 피해를 주었던 랜섬웨어는 점차 발전하여 파일을 유출하고 분산형 서비스 거부 공격과 같은 추가적인 공격을 수행하며, RaaS라는 새로운 수익 모델의 등장으로 전문 지식이 부족한 사람들도 랜섬웨어를 사용할 수 있게 되며 잠재적인 사용자가 증가하는 등 랜섬웨어의 위험성은 점점 높아지고 있다.

정보의 가치는 사회가 발전함에 따라 점차 증가하고 있다. 그만큼 귀중한 정보들을 대상으로 공격을 수행하는 랜섬웨어의 수요는 더욱 늘어날 것이고 공격 방식 역시 지금보다 정교해지고 강력해질 것이다. 또한, 일부 랜섬웨어 그룹의 경우는 국가의 지원을 받아 공격을 수행하여 이제는 개인과 기업의 문제로만 치부할 수 없을 것이다[11].

따라서 랜섬웨어로 인한 피해를 조금이나마 줄이기 위해서 다양한 랜섬웨어를 대상으로 분석 연구가 더 많이 수행될 필요가 있다.

본 논문에서는 Windows 환경에서의 Phobos 랜섬웨어의 동작과 암호화 파일에 대해 분석하였다. 분석 결과 랜섬 노트 발생 이후로는 삭제된 원본 파일을 복원할 수 없고 암호화키 역시 획득 불가능하였다는 한계가 존재하였다.

향후 연구에서는 5장 2절에서 제시하였던 간헐적 암호화로 남아있는 원본 데이터를 디스크 이미지에서 탐색하는 방법으로 원본 파일을 복구하는 방안을 Phobos 랜섬웨어와 유사하게 원본 파일에 파일을 덮어쓰지 않으면서 간헐적 암호화를 사용하는 랜섬웨어들을 대상으로 복호화 가능한 랜섬웨어가 존재하는지를 연구할 것이다.

참 고 문 헌

[1] 보안뉴스 “2024년 2분기 랜섬웨어 공격 18% 증가 ... 국내 1분기 대비 10배 폭증” https://m.boannews.com/html/detail.html?idx=132281

[2] Techcabal, “Nigerian cloud provider hit with ransomware attack as government agency works to “swiftly resolve incident””, https://techcabal.com/2024/07/10/cloud-providers-ransomware-attack/

[3] The Register, “Six ransomware gangs behind over 50% of 2024 attacks”, https://www.theregister.com/2024/08/13/lockbit_ransomware_stats/

[4] 차해성, 서승희, 이창훈, “역공학을 통한 Bianlian 랜섬웨어 복호화 방안 연구”, *디지털포렌식연구*, 17(3), 135-145, 2023

[5] 강수진, 이세훈, 김소람, 김대운, 김기문, 김종성, “키 재사용 공격을 통한 Ragnar Locker 랜섬웨어 감염 파일 복호화 및 활용 방안 연구”, *정보보호학회논문지*, 31(2), 221-231, 2021

[6] 이영주, “스트림 암호 기반 랜섬웨어에 대한 기술적 분석 동향”, *정보보호학회지*, 32(3), 49-56, 2022

[7] Luis Fernandez de Loaysa Babiano, Richard Macfarlane, Simon R. Davies, “Evaluation of live forensic techniques, towards Salsa20-Based cryptographic ransomware mitigation”, *Forensic*

Science International: Digital Investigation, Volume 46 301572, ISSN 2666-2817, 2023

[8] Karan Bhat Sumbly, Pradyuman K Kannan, Likhitha A Aralimara, Sushma E, "Static and Dynamic Analysis of Clop Ransomware", *2022 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*, pp. 48-52, 2022

[9] Fabrizio Cicala, Elisa Bertino, "Analysis of Encryption Key Generation in Modern Crypto Ransomware", *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 2, pp. 1239-1253, 2022

[10] GTT Korea, "서비스형 악성코드(MaaS)와 랜섬웨어(RaaS) 극성", https://www.gttkorea.com/news/articleView.html?idxno=8735

[11] 데일리시큐, "중국과 북한 해커들의 랜섬웨어 공격, 전 세계 주요 인프라 위협" https://www.dailysecu.com/news/articleView.html?idxno=157222

〈저 자 소 개〉

안 원 석 (Wonseok An)

학생회원

2018년 3월 : 한성대학교 IT융합공학부 재학

<관심분야> 정보보호, 디지털포렌식

박 명 서 (Myungseo Park)

종신회원

2013년 2월 : 국민대학교 수학과 이학사

2015년 2월 : 국민대학교 금융정보보안학과 이학석사

2014년 12월~2017년 2월 : 국가보안기술연구소 연구원

2021년 8월 : 국민대학교 금융정보보안학과 이학박사

2021년 9월~2022년 2월 : 국민대학교금융정보보안학과박사후연구원

2022년 3월~2023년 8월 : 강남대학교 ICT융합공학부 조교수

2023년 9월~현재 : 한성대학교 융합보안학과 조교수

<관심분야> 정보보호, 디지털포렌식