KARA
Korea Anti Ransomware Alliance

2023.11.

KARA 랜섬웨어 동향 보고서

KARA 랜섬웨어 동향 보고서

■ 랜섬웨어 트렌드

✓ 랜섬웨어 트렌드 분석

3분기에는 랜섬웨어 그룹의 압박 전략이 더욱 고도화되었다. 2분기에 종결된 줄 알았던 Clop 그룹의 데이터 유출 사태가 계속되었으며, 더 나아가 Clop 그룹은 ClearNet[1]과 토렌트를 통해 데이터 유출을 지속하고 있다. 이는 기존 Tor 브라우저의 다운로드 속도 문제를 보완하고 광범위한 접근성을 통해 피해 기업을 압박하려는 전략의 일환으로 볼 수 있다. 비슷한 방식으로 BlackCat 그룹에서는 피해 기업에 대한 세부 정보를 얻어올 수 있는 API를 제공하기 시작했다.

공격을 성공시키기 위한 전략과 공격 도구 또한 한 단계 발전했다. LockBit 그룹의 계열사는 랜섬웨를 감염시키기 위해 LockBit 랜섬웨어 뿐만 아니라 3AM으로 불리는 RaaS형 랜섬웨어를 동시에 준비하는 치밀한 전략을 세웠으며, LockBit 랜섬웨어가 보안 시스템에 의해 차단당하자 준비한 3AM 랜섬웨어를 사용하여 공격을 성공시킨 사례가 확인되었다. 반면 BlackCat 그룹은 Remcom[2]과 Impacket[3]이 내장된 Sphynx 랜섬웨어를 출시하였고, Abyss 그리고 Monti 그룹은 Linux 버전의 랜섬웨어를 선보였다. 랜섬웨어 그룹들은 다양한 운영체제를 타겟으로 하며 강력한 기능을 내장한 랜섬웨어를 사용하고, 때로는 여러 그룹의 랜섬웨어를 활용한 공격 전략을 사용하고 있다.

신규 랜섬웨어 및 그룹 활동

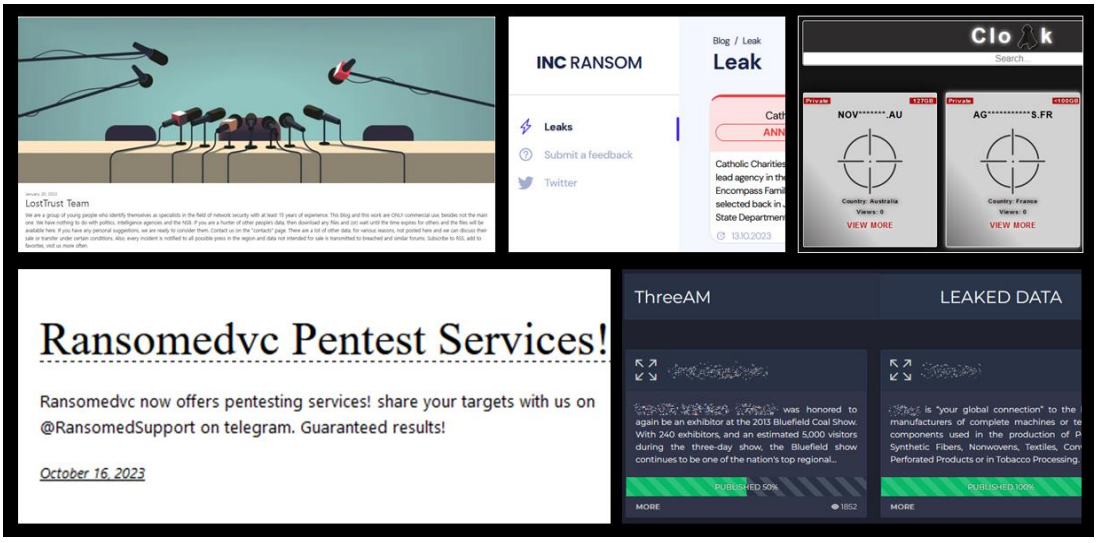


그림 1. 신규 랜섬웨어 및 그룹 활동

[1] ClearNet: 공개적으로 액세스할 수 있는 인터넷

[2] Remcom: 원격으로 PC제어가 가능한 도구로 PsExec의 오픈소스 버전

[3] Impacket: 네트워크 프로토콜과 서비스의 침투 테스트를 위한 툴 모음

3분기에는 CyClops, Knight, MetaEncryptor, LostTrust, UnderGround, Cactus, INC, Ransomed, Cloak, 3AM, CiphBit, CryptBB로 불리는 12개의 랜섬웨어 그룹이 발견되었다. 이 중 Cyclops 그룹은 7월부터 활동을 시작한 그룹으로, 7월 말에 계열사 패널과 랜섬웨어 업데이트를 예고한 뒤 Knight 그룹으로 리브랜딩[4]하여 재등장했다. 또한 LostTrust 그룹과 MetaEncryptor 그룹은 다크웹 유출 사이트에 동일한 이미지와 문구를 사용하고 있고, 22년 1분기에 발견된 Sfile2 랜섬웨어와 코드 유사성이 확인된다. Sfile2 랜섬웨어와 유사한 코드, LostTrust와 MetaEncryptor 랜섬웨어 그룹의 소개 문구까지 흡사한 점과 게시된 피해자가 모두 다르다는 점을 통해 단순 모방이 아님을 의미하며 LostTrust 그룹은 MetaEncryptor 그룹의 리브랜딩으로 추측할 수 있다.

일부 그룹들은 타 그룹과 연관성이 있거나 파트너십을 맺고 있는 것으로 확인되었다. 특히 8월에 발견된 Cloak 그룹은 GoodDay 그룹과의 연관성이 제기되고 있다. 이는 GoodDay 그룹이 사용하는 랜섬노트에서 Cloak 그룹이 사용하는 다크웹 주소가 기재되어 있었다는 근거를 토대로 한다. 더불어, Ransomed 그룹은 ClearNet인 Ransomed.vc 운영을 중단하고 다크웹으로 전환한 후 활동을 이어가고 있다. 이 과정에서 텔레그램 및 인터뷰를 통해 다크웹 포럼인 BreachForums와 랜섬웨어 그룹인 Stormous와 Everest 와의 파트너십을 강조했다. 이러한 조치는 해당 그룹들이 단순히 개인적 활동이 아니라, 다른 그룹들과 협력하며 보다 활발한 공격을 수행하기 위한 상생관계를 가지고 있는 것으로 볼 수 있다.

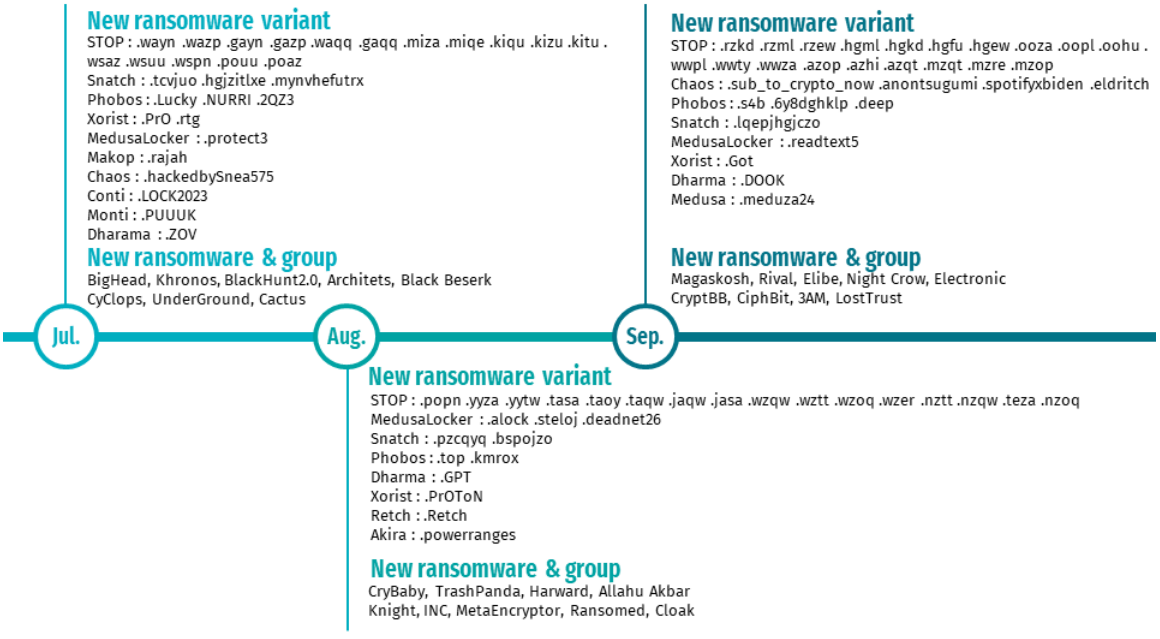


그림 2. 신규/변종 랜섬웨어 활동

[4] 리브랜딩: 랜섬웨어 공격자들이 운영을 공개적으로 중단한 후 새로운 이름으로 다시 운영하는 것

최근 LockBit 그룹의 계열사가 시스템을 공격하던 중 보안 시스템에 의해 LockBit 랜섬웨어가 차단되자 3AM 랜섬웨어를 활용하여 시스템을 감염시킨 사례가 발생했다. 이 사건을 통해 3AM 그룹의 존재가 드러났지만, 현재까지 3AM 그룹과 LockBit 그룹 간의 연관성은 확인되지 않았다. 이 사건에서 주목해야 할 점은 LockBit 그룹의 계열사가 공격을 성공시키기 위해 단순히 하나의 랜섬웨어가 아닌 다양한 RaaS[5]형 랜섬웨어를 보유하고, 선택적으로 활용했다는 점이다.

3AM 랜섬웨어는 Rust 언어로 제작되었는데, 이와 비슷하게 비주류 언어로 제작된 랜섬웨어들이 계속해서 등장하고 있다. 비주류 언어는 빠른 암호화 속도, 분석 및 탐지 우회, 크로스 플랫폼[6] 등의 이점이 있어 이를 채택하는 움직임이 지속되고 있는 것으로 보인다. Rust 언어로 제작된 SophosEncrypt는 정보 보안 업체 Sophos의 이름을 사칭하고 있으며, 일반적인 랜섬웨어 행위인 시스템 암호화뿐만 아니라 키 입력을 로깅하고 시스템을 원격으로 제어 할 수 있는 RAT의 기능 또한 포함하고 있다. 또한 8월에는 Nim언어로 제작된 Kanti 랜섬웨어가 발견되기도 했다.

랜섬웨어 공격 그룹 트렌드

Clop 그룹이 MOVEit 취약점을 이용한 대규모 공격 이후로, 다수의 기업에서 사용되는 소프트웨어의 취약점을 활용한 공격 사례가 늘어나고 있다. 특히 최근에는 LockBit 그룹과 Akira 그룹이 Cisco VPN 및 Firepower Threat Defense 취약점을 악용하여 기업들을 목표로 한 공격을 시도한 사례가 확인되었다. 이와 동시에, TripAdvisor 이메일로 위장한 Knight, Windows 보안 업데이트로 위장한 Magniber, 그리고 정상 소프트웨어 사이트 및 Google Ads를 통해 유포되는 BlackCat 등 사회공학 기법을 사용한 랜섬웨어 전파 사례도 증가하는 추세다. 이는 일부 그룹 또는 계열사간의 기술력 차이로 볼 수 있지만, 랜섬웨어 그룹이 최초로 설계한 전략을 쉽게 바꾸지 않는 것으로 볼 수 있다.

IAB[7]와의 협업을 통해 랜섬웨어 생태계는 한층 더 조직적이고 치밀해지고 있다. 서비스형 랜섬웨어 그룹은 계열사를 고용하고 초기 침투 경로를 IAB에게 구매하여 공격을 수행한 뒤 얻은 수익을 믹싱 서비스[8]를 통해 세탁하는 등 체계화된 양상을 띠고 있다. 이러한 변화로 인해 전문적인 지식이 없어도 랜섬웨어 공격이 가능하게 되어 피해 사례 역시 증가하고 있다. 또한 과거 랜섬웨어 그룹들은 데이터 암호화를 통해 몸값을 요구하는 것이 대부분이었지만, 복호화 툴이 공개되거나 탐지를 회피하기 위해 데이터 탈취만 수행해 몸값을 요구하는 그룹들이 하나둘씩 등장하고 있다.

---

[5] RaaS: 서비스형 랜섬웨어(Ransomware-as-a-Service)라는 의미로, 돈을 주고 랜섬웨어를 제공받는 서비스

[6] 크로스플랫폼: 하나의 언어와 도구로 다양한 플랫폼에서 동작이 가능한 형태

[7] IAB: Initial Access Broker의 약자로, 초기침투만을 전문적으로 수행하는 브로커

[8] 믹싱 서비스: 암호화폐 거래내역을 뒤섞는 서비스로, 가상자산 추적을 피하기 위해 사용

랜섬웨어 그룹활동 및 통계

3분기에는 지난 분기에 비해 8% 증가한 1,384건의 피해 사례가 확인되었다. 이는 Clop 및 Malas 그룹의 대규모 공격뿐만 아니라 8Base 그룹, NoEscape, LostTrust, Cactus 등의 신규 그룹들이 활발한 활동을 보이면서 더 많은 피해가 발생한 결과이다.

LockBit 그룹은 3분기에도 가장 많은 피해를 입혀 가장 큰 위협을 보였지만, 7월에는 Clop 그룹에서 대규모로 데이터를 게시하자 이를 의식이라도 한 듯 데이터를 게시하지 않다가, Clop의 활동이 줄어들자 다수의 데이터를 게시하는 모습을 보였다. 7월의 공격 수가 49건, 8월 공격수가 122건인 것을 감안하면 활동에 일부 변화가 있었다는 것을 알 수 있다. 또한 8월에는 그간 LockBit 그룹의 부실한 운영으로 인해 계열사가 불만을 표출하거나 이탈했던 사실이 드러났다. 데이터 게시 오류 및 다크웹 페이지 오류, 개발자 부재 및 신규 랜섬웨어 개발 지연 등으로 인한 신뢰성 하락이 가장 큰 이유인 것으로 확인됐다. 한편 LockBit은 8월과 9월에 걸쳐 2건의 국내 기업의 데이터를 게시했으며, MetaEncryptor, NoEscape 그룹에서도 각각 1건의 국내기업의 데이터를 게시한 이력이 있어 국내 기업의 피해 사례가 확인되었다.

최근 서비스형 랜섬웨어 그룹들은 멀티 플랫폼을 지원하여 다양한 플랫폼을 공격할 수 있도록 서비스를 제공하고 있으며, 취약점을 악용한 공격 사례가 지속 발견되고 있다. 취약점의 영향도에 따라 대규모 공격과 피해가 발생할 수 있어 랜섬웨어 그룹의 전략과 전술을 사전에 파악하여 능동적이고 선제적인 조치가 필요하다.

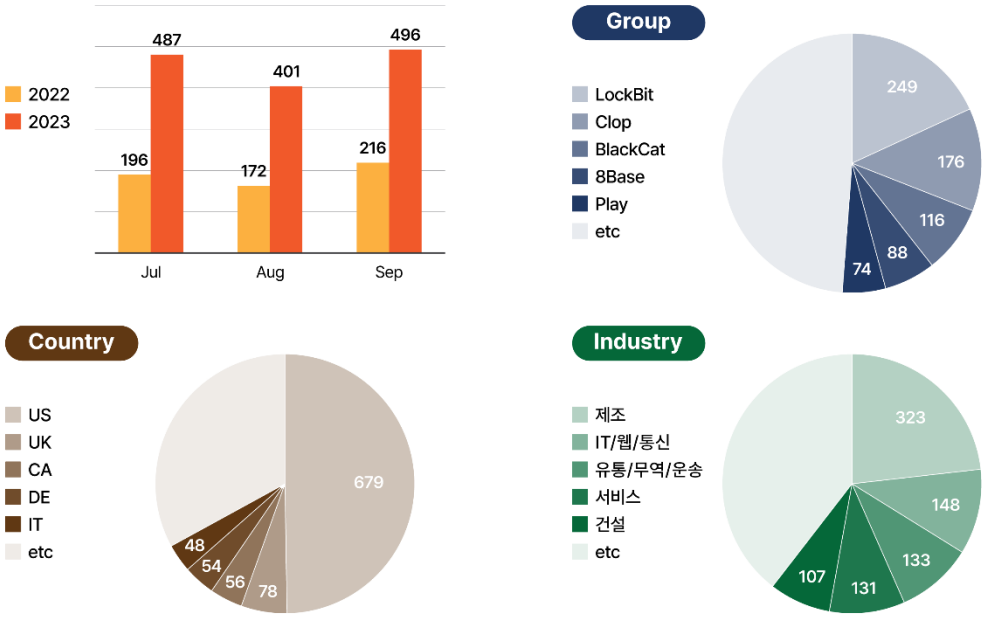


그림 3. 랜섬웨어 그룹 활동

✓ 복호화가 가능한 랜섬웨어 2종

3 분기에는 KeyGroup 에서 사용한 랜섬웨어에 대한 복호화 스크립트가 EclecticIQ 에 의해 공개되었다. 복호화가 가능한 샘플은 1 건으로, 해당 랜섬웨어는 파일 암호화에 고정된 키를 사용하여 파일 복호화가 가능했다. 또한 9 월에 발견된 Payola 랜섬웨어는 복호화 도구가 바이러스 토탈에 공개되어, 개인키와 공개키가 매칭되는 일부 랜섬웨어에 대해 복호화가 가능하다. 이와 같이, 암호화 키의 노출로 인한 실수나 암호화 알고리즘의 취약점, 암호화 알고리즘 대신 단순한 xor 연산 사용, 복호화 도구의 유출로 인한 개인키 유출 등으로 인해 복호화가 가능한 랜섬웨어 사례가 종종 확인되고 있다.

SK쉴더스 랜섬웨어 대응센터에서는 KeyGroup에서 사용한 PoliceRecords 랜섬웨어의 일부 변종 및 최근 RaaS형태로 운영되고 있는 NoBit 랜섬웨어에 대한 분석을 진행하였으며, 현재까지 확인된 일부 샘플에 대해서 복호화가 가능한것을 확인했다. 이에 해당 랜섬웨어에 대한 상세 분석 내용과 복호화 스크립트를 제공하고자 한다.

1. 유출된 빌더를 적극 활용한 KeyGroup 랜섬웨어

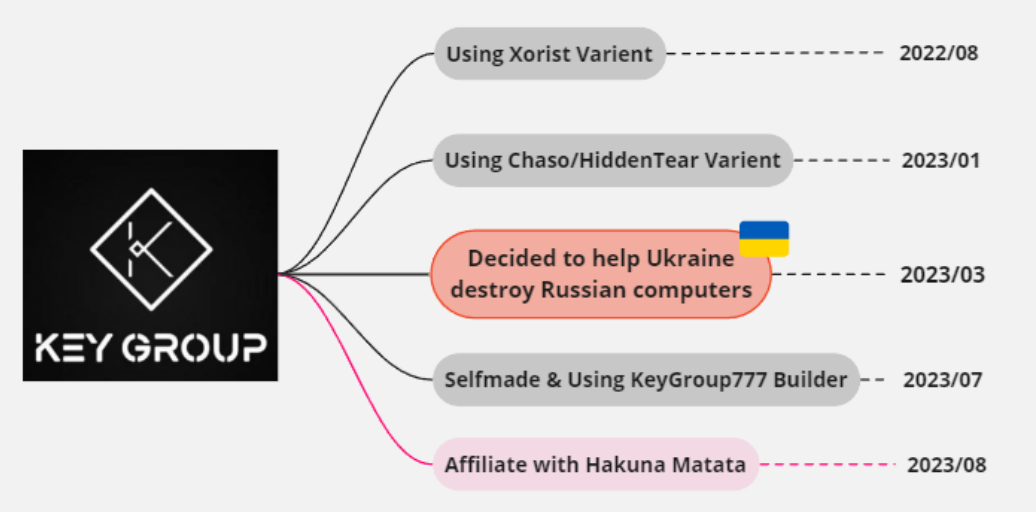


그림 4. KeyGroup 랜섬웨어 활동

KeyGroup 랜섬웨어는 2022년 8월에 처음 발견되었다. 초기 버전은 Xorist 랜섬웨어를 기반으로, 주로 러시아인을 대상으로 공격한 것으로 확인된다. 2023년 1월 초부터 Chaos 4.0 빌더[9]를 사용하여 변종을 유포하기 시작했고, 3월에는 랜섬노트를 통해 '우리는 우크라이나를 돕고 러시아의 컴퓨터를 파괴하겠다'라고 선언하여 반 러시아적인 성향을 드러냈다. 4월부터는 DarkStore라는 다크웹 포럼에서 활동을 시작하며, NjRAT[10] 악성 코드의 유포와 러시아인의 텔레그램 채널을 공격하는 등의 활동을 이어가고 있다.

한편 이들은 올해 4월부터 텔레그램을 통해 지속적으로 본인들이 제작한 랜섬웨어 빌더 뿐만 아니라 과거에 유출된 Chaos, HiddenTear, GoldenEye 등 다양한 랜섬웨어 빌더 및 소스코드를 텔레그램으로 공유해온 정황과, Anabelle, RuRansom, Cyborg 랜섬웨어를 사용한 정황이 확인 됐다. 또한 8월에는 HakunaMatata 랜섬웨어와 제휴 중이라는 사실을 밝히기도 했다. 최근 이와 같이 유출된 랜섬웨어 빌더나 소스코드를 사용하는 그룹이 늘어나는 추세이며, 최근에는 HelloKitty 랜섬웨어의 소스코드가 유출된 만큼 해당 소스코드를 사용한 변종이 증가할 것으로 추측된다.

SK쉴더스 랜섬웨어 대응센터는 KeyGroup에서 사용한 랜섬웨어를 분석하고, 일부 샘플에서 복호화가 가능함을 확인했다. 추가로, 복호화가 가능한 랜섬웨어는 2022년 5월부터 발견된 PoliceRecords 랜섬웨어의 변종으로 해당 랜섬웨어의 변종 일부에서도 복호화가 가능함을 확인했다. 이에 해당 분석 내용과 복호화 스크립트를 제공하고자 한다.

---

[9] 빌더(Builder): 랜섬웨어를 만들거나 배포할수 있는 도구

[10] NjRAT: Remote Access Trojan의 일종으로, 공격자의 명령을 받아 다양한 악성 행위를 수행하는 악성코드

1) 특징

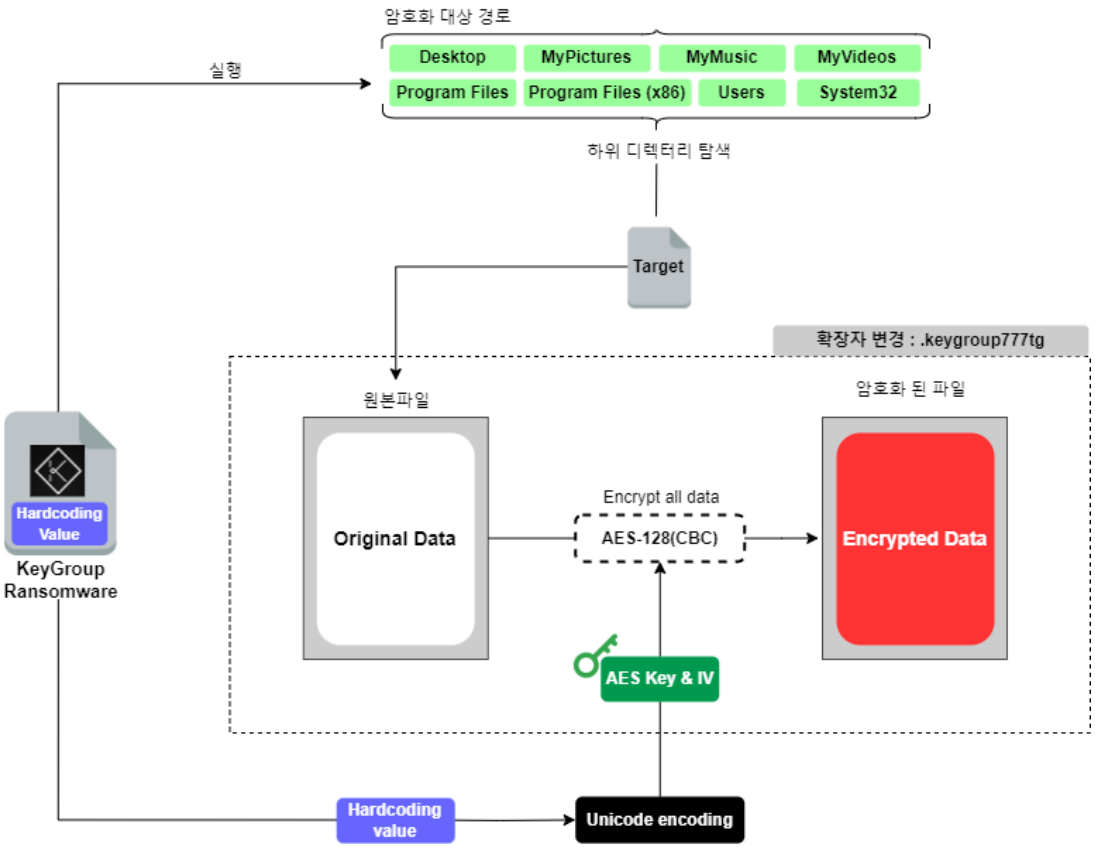


그림 5. KeyGroup 랜섬웨어 특징

- 해당 랜섬웨어는 2023 년 9 월 14 일에 발견되었으며, KeyGroup 에서 사용한 랜섬웨어로 2022 년 5 월경에 발견된 PoliceRecords 랜섬웨어의 변종으로 확인된다.
- Desktop, MyPictures, MyMusic, MyVideos, Program Files, Program Files(x86) 등 지정된 폴더에 포함되어 있는 파일들을 모두 암호화 하며, 암호화 된 파일에 .keygroup777tg 확장자가 추가된다
- 파일을 암호화 하기 위해 AES-128(CBC) 알고리즘을 사용한다. 이때 사용되는 키는 내부에 저장되어 있는 고정 값에 유니코드 인코딩을 적용해 생성되며, 키와 초기벡터는 동일한 값을 사용한다.
- 암호화를 수행한 뒤 SOFTWARE₩Microsoft₩Windows₩CurrentVersion₩Policies₩System 레지스트리에서 FilterAdministratorToken, EnableLUA, DisableTaskMgr, DisableRegistryTools 값을 변경해 UAC(사용자 계정 컨트롤), 작업관리자, 레지스트리 변경 기능에 접근할 수 없도록 한다.

2) 복호화 방안

KeyGroup 랜섬웨어는 고정된 값에 유니코드 인코딩을 적용해 AES-128 알고리즘에 사용할 키와 초기벡터를 생성한다. 일반적으로 완성도가 높은 랜섬웨어는 파일마다 랜덤한 키와 초기벡터값을 생성해 파일 암호화에 사용하고, 비대칭키 알고리즘을 사용해 키를 보호하는 하이브리드 암호화 기법을 사용하지만, 해당 랜섬웨어에는 고정된 값을 사용해 키 값을 생성한 뒤 모든 파일을 암호화 했으며, 키를 보호하지 않아 복호화가 가능했다.

발견 일시	키 값 암호화 여부	변종
23/08/30	O	Xorist
23/01/06 ~ 23/06/24	O	HiddenTear/Chaos
23/08/03	X	HiddenTear/Chaos
23/08/28	O	Anabelle
23/09/07	O	RURansom
23/09/14	X	PoliceRecords
23/09/19	O	Cyborg
23/09/25	O	UxCryptor

표 1. KeyGroup 사용 랜섬웨어

[표 1]을 보면 KeyGroup 에서 사용한 랜섬웨어 대부분이 하이브리드 암호화 기법을 통해 암호화에 사용된 키가 보호되었지만 8 월 3 일, 9 월 14 일에 발견된 샘플에서만 키가 보호되지 않는 것으로 보아 공격자가 랜섬웨어를 제작하는 과정에서 실수를 한 것으로 추측된다. [표 1]에서 8 월 3 일 발견된 샘플은 Eclecticiq 에서 스크립트를 공개한 사례가 있다.

SK 쉴더스 랜섬웨어 대응센터에서 확인한 복호화 가능한 샘플은 9 월 14 일에 발견된 샘플이다. 해당 KeyGroup 랜섬웨어는 2022 년 5 월경 발견된 PoliceRecords 랜섬웨어의 변종으로 확인됐으며 PoliceRecords 랜섬웨어 변종 일부에서도 복호화가 가능함을 확인했다.

파일명	MD5
Police_Records.exe	00d77230603c745c638c5de737d1593e
Police_Records.exe	d7a7df59b8979b97d547972b307a4740
RubberDucky.exe	7C3EADFECFE56137704664A9CBED3544

표 2. 복호화 가능한 랜섬웨어

만일 암호화된 파일의 확장자가 .CRYPT, .keygroup777tg 또는 랜섬노트의 이름이 FAQ.txt, Rubber_Decrypt0r.txt 일 경우 아래의 스크립트를 통해 복호화를 시도해 볼 수 있으며, [표 2] 와 같을 경우 아래의 스크립트를 통해 복호화가 가능하다.

```
import pip
import os
import subprocess

print("Download moudles for decrypt")
pip.main(['install','requests'])
pip.main(['install','pycryptodome'])
pip.main(['install','dnfile'])
pip.main(['install','dncil'])
import requests
from Crypto.Cipher import AES
from Crypto.Util.Padding import unpad

# Download Script(Open source by Mandiant)
def DownloadScript():
    url =
'https://raw.githubusercontent.com/mandiant/dncil/main/scripts/print_cil_from_dn_file.py'
    res = requests.get(url)
    with open('print_cil_from_dn_file.py','w') as f:
        f.write(res.text)

def ExecuteScript(RansomwareFileName):
    result = subprocess.run(f'print_cil_from_dn_file.py {RansomwareFileName} >>
dis.txt',shell=True)

def GetRansomwareFileName():
    RansomwarefileName = input('Input the ransomware sample file name(Full path): ')
    return RansomwarefileName

def FindKey():
    file_path = "dis.txt"
    target_string_1 = "Method: EncryptFile"
    target_string_2 = "nop"
    target_string_3 = "ldstr"

    try:
        with open(file_path, "r") as file:
            lines = file.readlines()
            for i in range(len(lines)):
                if target_string_1 in lines[i]:
                    if target_string_2 in lines[i+1] and target_string_2 in lines[i+2]:
                        if target_string_3 in lines[i+3]:
                            keysource_line = lines[i+3].strip()
                            keysource = keysource_line.split('"')[1]
```

```
                        print("keysource is exist :", keysource)
                        break

    except FileNotFoundError:
        print(f"Cannot found '{file_path}'.")
        exit()
    except Exception as e:
        print(f"Error : {e}")
        exit()

    # Check key length
    if len(keysource)==0:
        print("Key not found")
        exit()

    elif len(keysource)!=8 & len(keysource)!=16 :
        print("Key length is incorrect")
        exit()

    # Unicode encode
    array = bytearray()
    for i in keysource:
        array.append(ord(i))
        array.append(0x00)

    # Get key, IV
    key = array
    iv = array
    print('Key : %s IV : %s'%(key,iv))

    return key,iv

# AES decrypt func
def DecryptFile(file_path,key,iv):

    try:
        with open(file_path, 'rb') as file:
            ciphertext = file.read()
            cipher = AES.new(key, AES.MODE_CBC, iv)
            decrypted_data = unpad(cipher.decrypt(ciphertext), AES.block_size)
        with open(file_path, 'wb') as file:
            file.write(decrypted_data)
        print(f'Decrypted: {file_path}')
```

```
        # Remove encrypted extensions
        decrypted_file_path = os.path.splitext(file_path)[0]

        # Verify that the same file already exists
        while os.path.exists(decrypted_file_path):
            decrypted_file_path += '_copy'
        os.rename(file_path, decrypted_file_path)
        print(f'Renamed to: {decrypted_file_path}')

    except Exception as e:
        print(f'Error decrypting {file_path}: {e}')

# Travel directory
def TravelDirectory(Directory,Extension,key,iv):
    for foldername, subfolders, filenames in os.walk(Directory):
        for filename in filenames:
            if filename.endswith(Extension):
                file_path = os.path.join(foldername, filename)
                if os.path.exists(file_path) and os.path.getsize(file_path)==0:
                    os.remove(file_path)
                    continue
                DecryptFile(file_path,key,iv)

def GetExtension():
    Extension = input('Input the encrypted extension(without comma): ')
    return '.' + Extension

def GetStartDirectory():
    start_directory = input('Input the start directory(without :\\\\): ')
    return start_directory+':\\'

if __name__ == "__main__":
    DownloadScript()
    ExecuteScript(GetRansomwareFileName())
    key, iv = FindKey()
    TravelDirectory(GetStartDirectory(),GetExtension(),key,iv)
```

스크립트 1. 복호화 스크립트

3) IoC

MD5	00d77230603c745c638c5de737d1593e d7a7df59b8979b97d547972b307a4740 7C3EADFECFE56137704664A9CBED3544
File name	Police_Records.exe RubberDucky.exe

2. 신규 RaaS, NoBit 랜섬웨어

그림 6. 최근 홍보되고 있는 RaaS 랜섬웨어

보이지 않는 곳에서 RaaS 랜섬웨어가 우후죽순 생겨나고 있다. 이들은 주로 텔레그램과 다크웹에서 활동하며 회원을 모집하고 수익을 창출한다. 7월에는 NoBit이라는 RaaS 형태의 랜섬웨어가 텔레그램과 다크웹에 등장했는데, 이들은 NoBit 랜섬웨어를 차세대 랜섬웨어로 소개하며 빠른 암호화, 타깃 PC에서 사용중인 언어(한국어, 영어 등) 탐지 기능, 사용의 용이성 등과 같은 기능을 강조하며 광고하고 있다. 하지만 기존의 RaaS형 랜섬웨어와는 별반 차이가 없으며 해시가 노출된 일부 샘플에서는 복호화가 가능해 차세대 랜섬웨어라는 말에 의구심이 든다. 현재 NoBit 랜섬웨어 빌더는 $200(한화 27만원)에 판매되고 있으며 랜섬웨어의 전체 소스코드는 $1000(한화 135만원) 수준으로 거래되고 있다.

NoBit 랜섬웨어 빌더는 AES-128 키 값 설정, 랜섬노트 수정, 몸값 표시 금액 변경, 특정 언어 사용 시 암호화 중지, 암호화할 확장자 지정 등 다양한 기능을 제공하여 공격자가 쉽게 맞춤 설정할 수 있는 장점을 가지고 있다. 이러한 RaaS 형태의 랜섬웨어는 해킹에 대한 전문 지식이 없는 사람들도 단순히 금액을 지불함으로써 사용할 수 있다는 점에서 큰 문제점을 안고 있다.

1) 특징

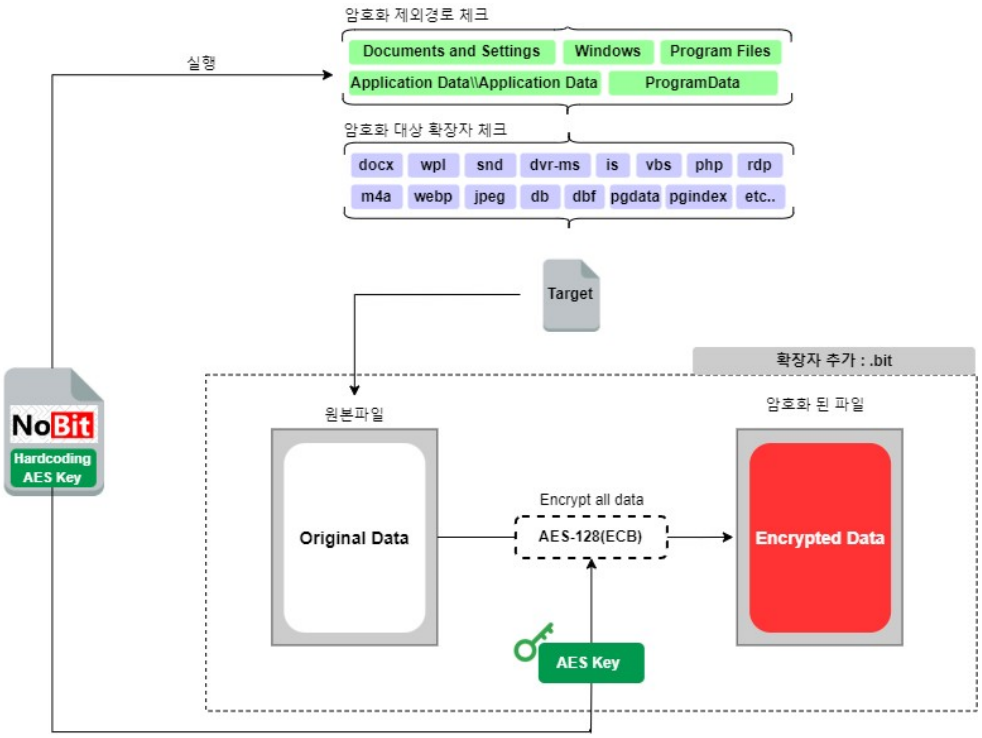


그림 7. NoBit 랜섬웨어 특징

- 시스템 복구를 방해하기 위한 목적으로 VSS[11]를 삭제한다.
- Documents and Settings, Windows, Program Files, ProgramData 등 PC 실행에 관련된 디렉터리일 경우에는 파일을 암호화 하지 않는다. 또한 파일의 확장자를 검사해 docx, wpl, snd, vbs 등 지정된 확장자와 일치할 경우에만 파일을 암호화한다.
- 데이터 암호화에는 AES-128(ECB모드) 알고리즘을 사용하며, 이때 사용되는 키 값은 랜섬웨어 내부에 하드코딩 되어 있다. 암호화 된 파일은 기존 파일명 뒤에 .bit 확장자가 추가된다.
- 파일 암호화가 끝나면 랜섬웨어 내부에 저장되어 있는 bitmap 파일을 사용해 바탕화면을 변경한다. 이후 Desktop(바탕화면) 경로에 Decryptor.exe 파일을 생성하고 실행한다. 해당 파일은 랜섬노트의 내용을 표시하며 암호화에 사용된 Key를 입력 받아 파일을 복호화 하는 기능을 가지고 있다.
- %Temp% 경로에 destruct.bat 파일을 생성한 뒤 실행한다. 해당 배치파일은 랜섬웨어를 삭제하는 기능을 가지고 있다.

---

[11] VSS: Volume Shadow Copy의 약자로, 특정 시점의 복사본 또는 스냅샷

2) 복호화 방안

NoBit 랜섬웨어는 파일 내부에 저장되어있는 128비트 키를 사용해 AES 알고리즘으로 데이터를 암호화한다. 이 때 파일마다 암호화에 사용되는 키가 동일하며 해당 키 값을 보호하지않아 복호화가 가능하다. 다만 NoBit 랜섬웨어는 파일 암호화를 수행한 뒤에 자가 삭제하기 때문에, 침해사고 조사를 통해 해시 혹은 샘플 확보가 필요하다. 해시가 아래의 표와 같은 경우 매칭되는 키 값으로 파일 복호화가 가능하며, 해시가 테이블에 없는 경우에는 확보한 샘플의 리소스 데이터에 존재하는 키 값을 통해 파일 복호화가 가능하다.

파일명	MD5	키 값(텍스트)
Botnet Virus Remover.exe	cad2d5524d0f66bb1017e206d28d2452	FF2934E360B2F1E6
try.exe	27c00e46185d476e33961c676f07774c	F60074D9D3F5EA5E
232464727	b10033b91ab6d47547871dfe361272bb	6CDF3A165152908D

표 3. NoBit 랜섬웨어

만일 NoBit 랜섬웨어에 감염되었을 경우, 공격에 사용된 랜섬웨어가 [표 3] 해시와 일치하는지 확인하거나 실제 랜섬웨어 샘플을 확보하여야 한다. 만일 [표 3] 해시와 일치하는 경우, 복호화 스크립트를 실행시키고 해시 값에 대응하는 키 값을 입력해 복호화를 수행하면 된다.

샘플의 해시가 [표 3]에 존재하지 않을 경우, JetBrains사 DotPeek 또는 오픈소스 도구 DnSpy 등의 .NET 디컴파일 도구를 사용해야한다. NoBit 랜섬웨어는 암호화에 사용된 키를 "key" 라는 이름의 리소스 형태로 저장되어 있어 해당 리소스 데이터를 찾아 키 값을 확보한 뒤 복호화 스크립트를 실행시켜 복호화를 수행할 수 있다.

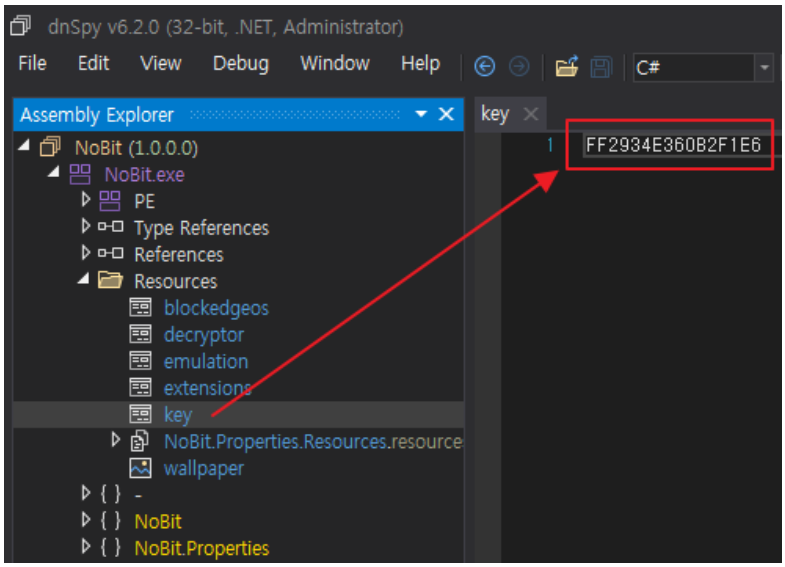


그림 8. Nobit 랜섬웨어 키 경로

```python
import os
import pip
import winreg
pip.main(['install','pycryptodome'])
from Crypto.Cipher import AES
from Crypto.Util.Padding import unpad

def SetWallpaper():
    regpath = r"Control Panel\Desktop"
    name = "Wallpaper"
    value = r"c:\windows\web\wallpaper\windows\img0.jpg"
    try:
        regkey = winreg.OpenKey(winreg.HKEY_CURRENT_USER, regpath, 0,
                                winreg.KEY_SET_VALUE)
        winreg.SetValueEx(regkey, name, 0, winreg.REG_SZ, value)
        winreg.CloseKey(regkey)
        return True
    except WindowsError:
        return False

# Get AES Key
def GetKey():
    key = bytes(input('Input the AES Key(in text) : '),'utf-8')
    return key

# AES decrypt
def decrypt_aes(ciphertext, key):
    cipher = AES.new(key, AES.MODE_ECB)
    plaintext = unpad(cipher.decrypt(ciphertext), AES.block_size)
    return plaintext

# Find and decrypt .bit files in a directory
def decrypt_bit_files(directory,key):
    for foldername, subfolders, filenames in os.walk(directory):
        for filename in filenames:
            if filename.endswith('.bit'):
                file_path = os.path.join(foldername, filename)
                with open(file_path, 'rb') as file:
                    encrypted_data = file.read()
                    decrypted_data = decrypt_aes(encrypted_data, key)
                # Create a filenmae with the .bit extension removed
                new_filename = os.path.splitext(filename)[0]
                new_file_path = os.path.join(foldername, new_filename)
                with open(new_file_path, 'wb') as file:
```

```
                    file.write(decrypted_data)
                # Delete an existing .bit file
                os.remove(file_path)
                print(f'Decrypted and renamed: {new_file_path}')

# Setting the start directory
def GetStartDirectory():
    start_directory = input('Input the start directory(without :\\\\) : ')
    return start_directory + ':\\'

# Main function
if __name__ == "__main__":
    decrypt_bit_files(GetStartDirectory(),GetKey())
    SetWallpaper()
    print('decrypt done')
```

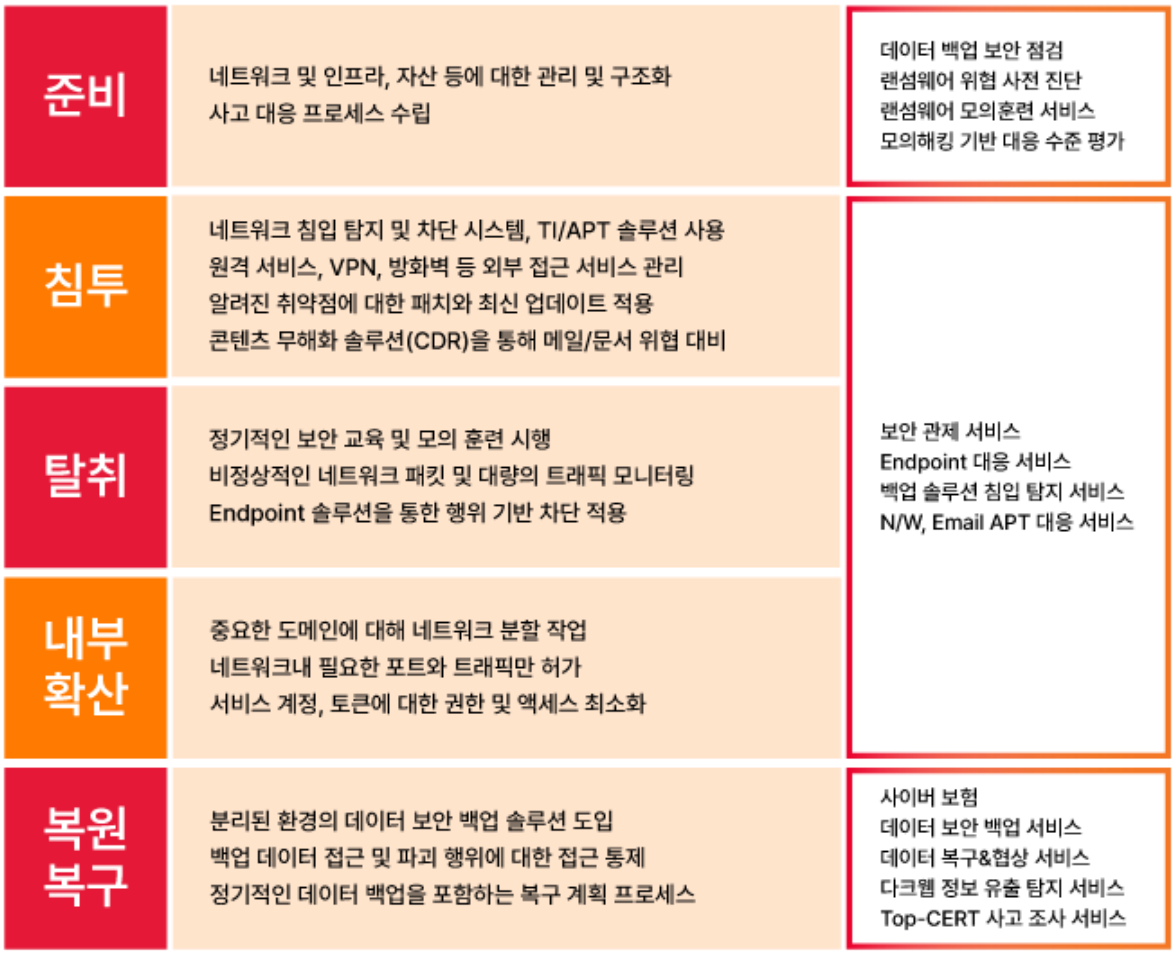
스크립트 2. 복호화 스크립트

3) IoC

MD5	cad2d5524d0f66bb1017e206d28d2452 27c00e46185d476e33961c676f07774c b10033b91ab6d47547871dfe361272bb
File name	Botnet Virus Remover.exe try.exe 232464727

■ 랜섬웨어 Mitigations

SK 쉴더스 랜섬웨어 대응센터는 3 분기 랜섬웨어 동향보고서를 통해 복호화 가능한 KeyGroup, NoBit 랜섬웨어 2 종에 대해 상세히 분석하고, 복호화 스크립트를 배포했다. 해당 랜섬웨어들은 파일 내부에 하드코딩된 키 값을 사용해 파일을 암호화하고, 키를 보호하지 않아 샘플이나 샘플의 해시가 존재하는 경우에 한해 파일 복호화가 가능하다. 하지만 랜섬웨어 그룹들은 파일 암호화 뿐만 아니라 데이터를 탈취해 이를 빌미로 돈을 요구하는 이중협박 전략을 사용하고 있기 때문에 피해를 예방하기 위해 타깃형 APT 공격에 대한 대비와 침입에 대한 각 단계별 적절한 보안 요소 및 프로세스를 마련하여 공격자 그룹이 목표를 달성하기 전에 탐지하고 차단할 필요가 있다.



안녕을 지키는 기술 | SK 쉴더스

SK쉴더스㈜ 13486 경기도 성남시 분당구 판교로227번길 23, 4&5층
https://www.skshieldus.com

발행인 : SK쉴더스 EQST사업그룹 & KARA(Korea Anti Ransomware Alliance)
제　작 : SK쉴더스 커뮤니케이션그룹