**Một số phương pháp trích xuất malicious AutoIT script**

## 1. Sơ lược về AutoIT

AutoIt là một ngôn ngữ kịch bản miễn phí được thiết kế nhằm tự động hóa các nhu cầu cơ bản của người dùng cũng như các tác vụ liên quan tới GUI Windows. AutoIt thường rất nhỏ và có khả năng thực thi trên tất cả các hệ điều hành Windows mà không đòi hỏi các thư viện "runtimes" như các chương trình được lập trình bằng Visual C++, ... Ngôn ngữ này được sử dụng cho các mục đích như:

- Mô phỏng lại các tổ hợp phím, thao tác di chuyển chuột và các thao tác điều khiển cửa sổ.
- Nhận thông tin và tương tác với các thành phần điều khiển như edit boxes, check boxes, list boxes, button, status bar.
- Tương tác với các tiến trình.
- Gọi trực tiếp tới các DLL và các hàm Windows API.
- Các script có thể được biên dịch thành các tập tin thực thi độc lập.
- ...

Nhờ vào những tính năng tiện lợi ở trên mà việc sử dụng AutoIT để viết phần mềm độc hại không phải là mới. Nó có thể được sử dụng làm keylogger, downloader, hoặc giai đoạn thăm dò trong quá trình lây nhiễm máy tính của nạn nhân. Tại Việt Nam, mã độc dạng này đã xuất hiện từ những năm 2006, điển hình trong đó là mã độc XRobots.

## 2. Công cụ sử dụng

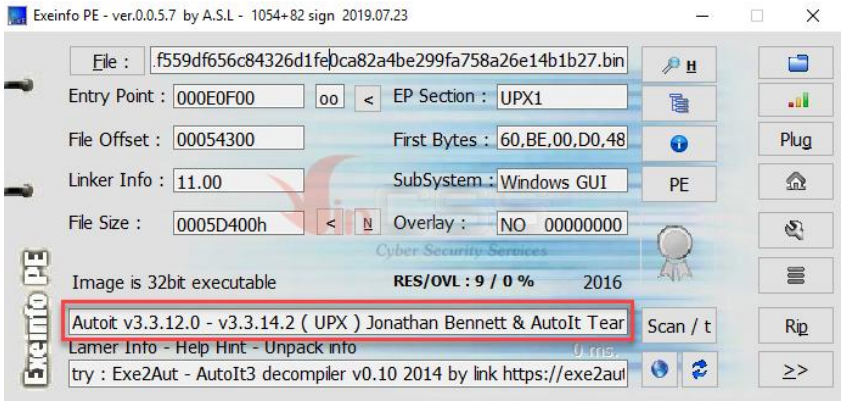
Danh sách các công cụ được sử dụng trong bài viết:

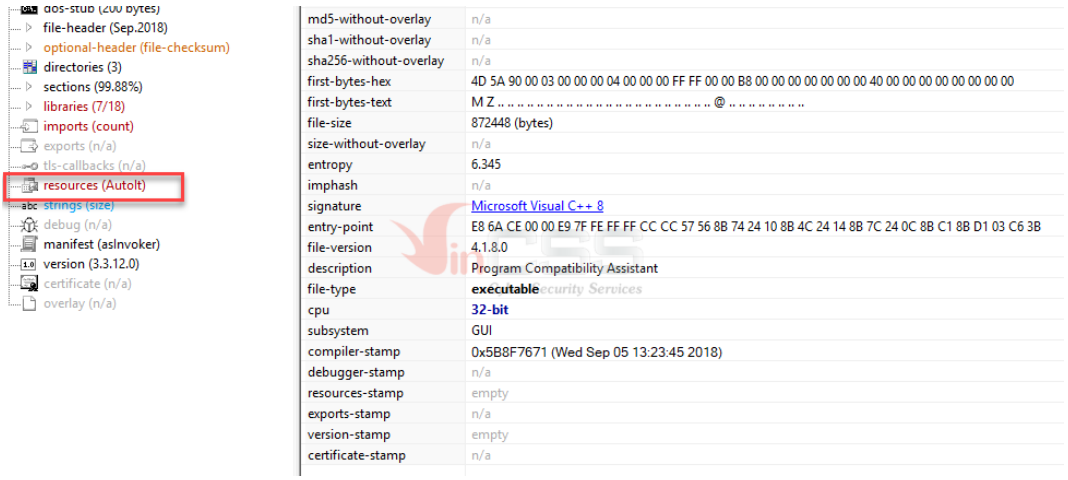
- Exeinfo PE/ PEStudio
- Hex Editor (HxD)
- Resource Hacker
- Aut2Exe v3 Converter
- Exe2Aut Decompiler
- autoit64to32.py & AutoItSC.bin (autoit-v3.2.8.1)

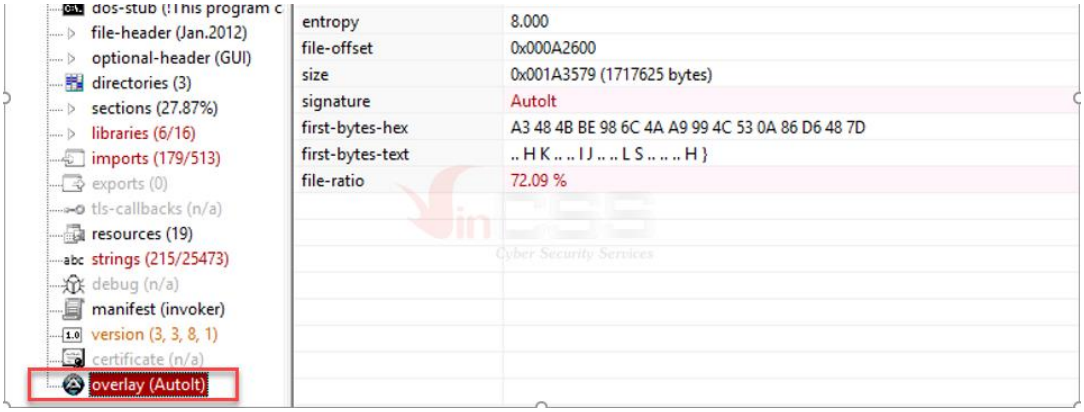
## 3. Cách nhận biết AutoIT

Khi phân tích một tập tin nghi ngờ là mã độc, thường người phân tích sẽ sử dụng các công cụ để kiểm tra xem tập tin đó có định dạng gì, được biên dịch bằng ngôn ngữ nào, có bị packed hay không, ... Với AutoIt sẽ có các dấu hiệu nhận biết như sau.

- Thông qua các công cụ như **Exeinfo PE** hoặc **PEStudio**:

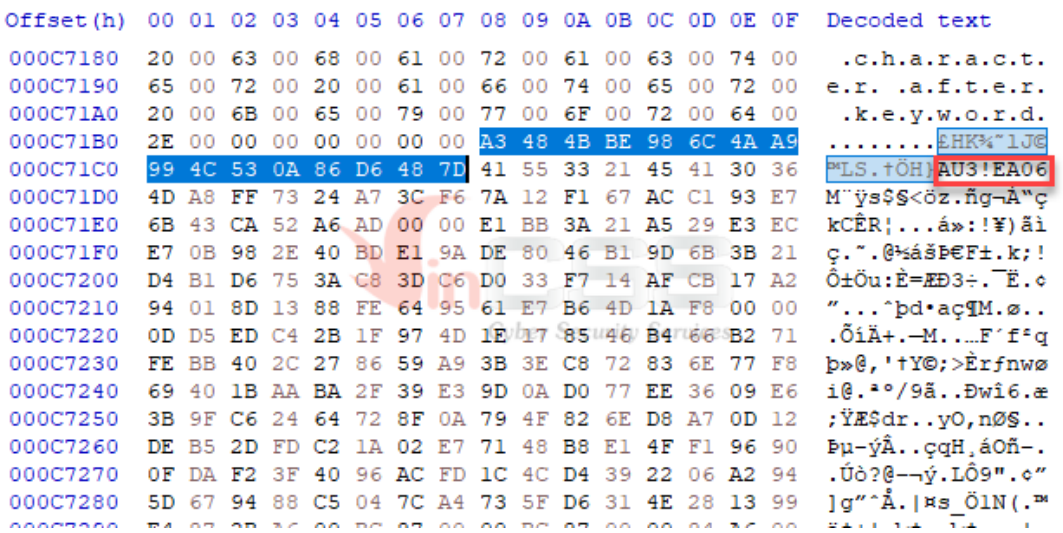





♦ Thông qua Magic number: *A3484BBE986C4AA9994C530A86D6487D*

Khi biên dịch một AutoIt script với công cụ **Aut2Exe**, theo mặc định, một tệp thực thi pack bằng UPX được tạo ra. Mở bằng một trình Hex editor và tìm kiếm chuỗi magic number nói trên, theo sau sẽ là dấu hiệu nhận dạng của AutoIt thông qua chuỗi **AU3!EA06**.

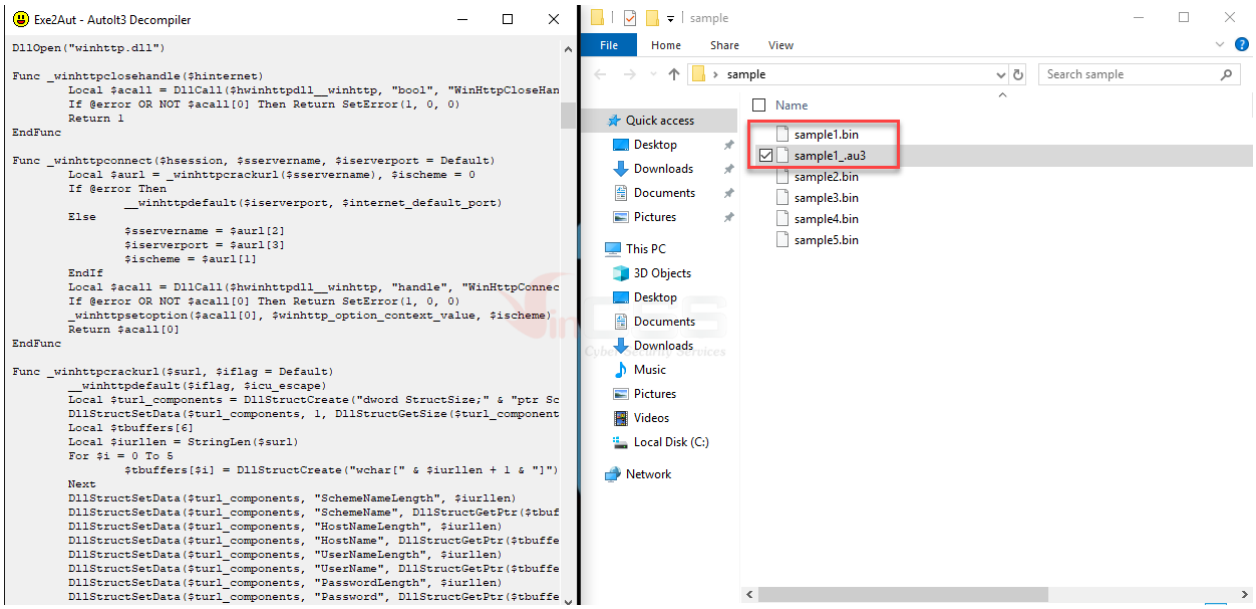



## 4. Phương pháp trích xuất script

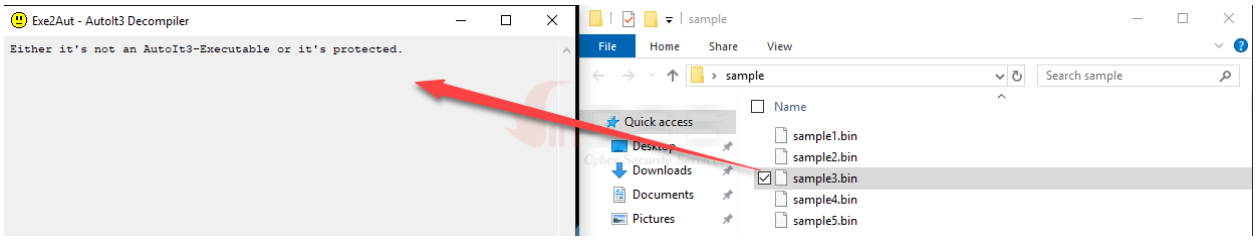
Các phương pháp trình bày dưới đây được tham khảo từ các chuyên gia @VK_Intel; @0xffff0800; @Hexacorn

♦ Phương pháp thứ nhất: "Drag & Drop"

Sở dĩ gọi phương pháp này là "drag & drop" là vì nó rất đơn giản, chỉ việc kéo thả một AutoIt exe-file vào trình **Exe2Aut** để nhận được script gốc. **Exe2Aut** sẽ tự động lưu script tại cùng thư mục với phần mở rộng là <filename>_.au3:

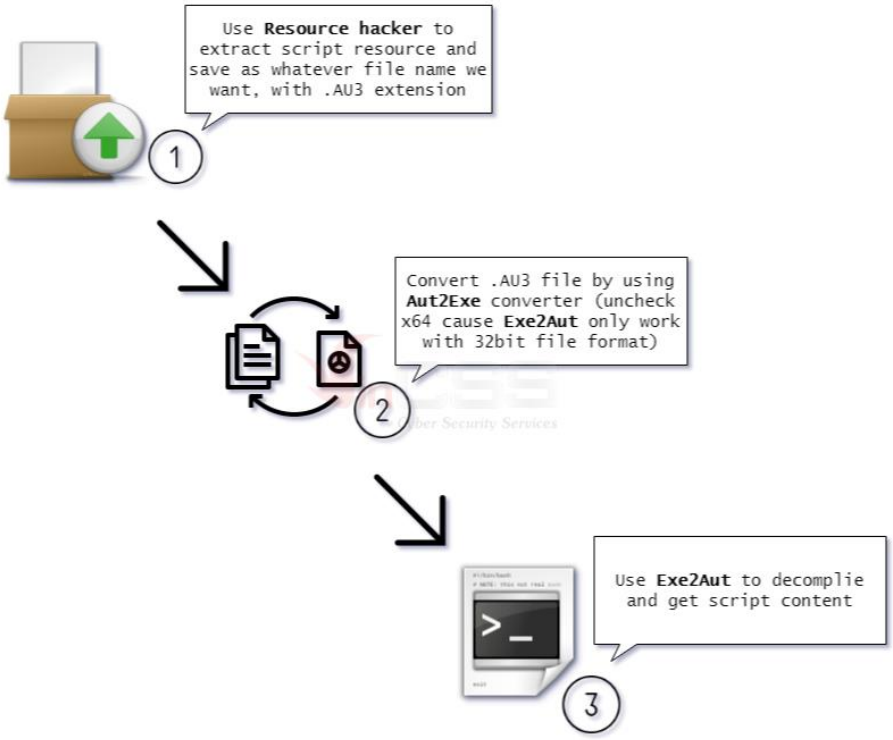



Tuy nhiên, không phải trường hợp nào **Exe2Aut** cũng hoạt động, ví dụ như trường hợp này:




♦ Phương pháp thứ hai

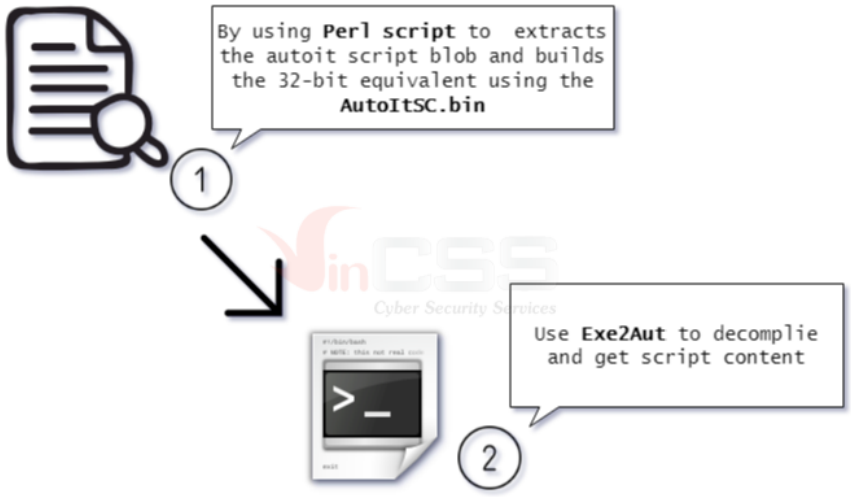
Để khắc phục cho trường hợp không trích xuất được bằng phương pháp đầu, phương pháp này sử dụng **Resource Hacker** để trích xuất script resource, lưu lại với tên bất kì với phần mở rộng là .au3. Tiếp theo, chuyển đổi file đã lưu sang định dạng PE bằng công cụ **Aut2Exe**. Lưu ý bỏ tùy chọn "*Compile for system x64*" trong quá trình thực hiện vì **Exe2Aut** không hỗ trợ x64 executables.




Dưới đây là phần demo minh họa cho phương pháp này

♦ Phương pháp thứ 3

Sử dụng Python script (được convert từ autoit64to32.pl (Perl script) của @Hexacorn) để trích xuất AutoIT script blob từ file thực thi và chuyển đổi sang định dạng 32 bit tương ứng bằng cách sử dùng **AutoItSC.bin** (32 bit) đi kèm trong bộ **autoit-v3.2.8.1**. File sau chuyển đổi sẽ thông qua **Exe2Aut** để lấy ra nội dung của script:




Nội dung của **autoit64to32.py** như sau:

```
use strict;
use warnings;

my $f=shift || die ("Gimme a file name!");

print STDERR "Processing '$f':\n";
print STDERR "- Reading 'AutoItSC.bin'\n";
open F,"<AutoItSC.bin";
binmode F;
read F,my $a, -s 'AutoItSC.bin';
close F;

print STDERR "- Reading '$f'\n";
open F,"<$f";
binmode F;
read F,my $d, -s $f;
close F;

print STDERR "- Looking for the script\n";
if ($d=~/\xA3\x48\x4B\xBE\x98\x6C\x4A\xA9\x99\x4C\x53\x0A\x86\xD6\x48\x7D/sg)
{
   my $pd=(pos $d)-16;
   print STDERR "- Script found @ ".sprintf("%08lX",$pd)."\n";
   print STDERR "- Creating 32-bit version '$f.a32.exe'\n";
   open F,">$f.a32.exe";
   binmode F;
   print F $a.substr($d,$pd,length($d)-$pd);
   close F;
}
else
{
   print STDERR "- Script not found !\n";
}
```

Dưới đây là phần demo minh họa cho phương pháp này: