Detecting ZeuS | eternal-todo.com

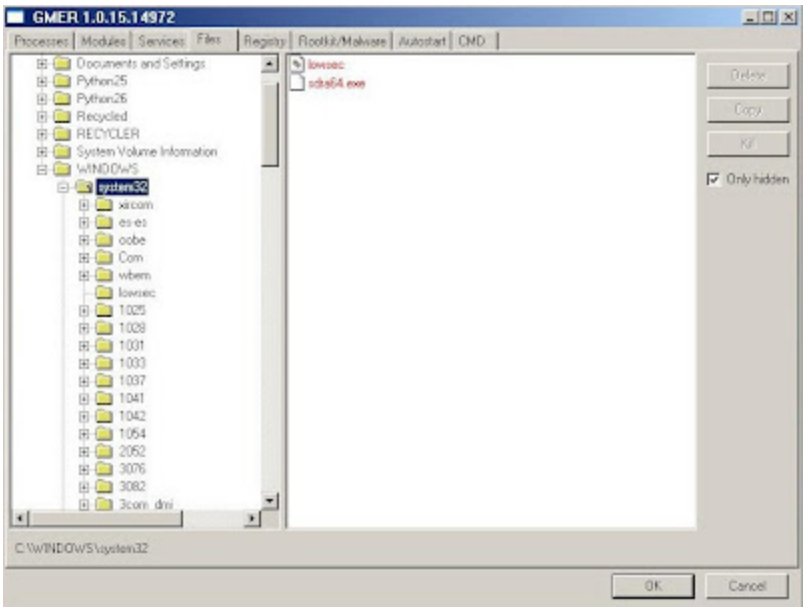
Detecting ZeuS

- Botnet
- Detection
- Malware
- ZeuS

In the S21sec blog we have been talking some time ago about our dear friend, almost one more colleague: ZeuS. It is a malware with more than 3 years of life which continues changing and evolving to hide itself better and making the fraud more efficient. But what we maybe have not mentioned yet is how to know if our little friend is here, spying all our movements and reporting all of this to its parents, because sometimes the AV software is not so effective as we expect.

There are several evidences in its different versions which mean that we are infected with ZeuS:

- Filesystem
  ZeuS leaves a trace in the filesystem when it's installed in the computer, but it hides and blocks all the files it creates, avoiding that a normal user can see and delete them. The solution to find these files is using antirootkit software which will show us the hidden files.



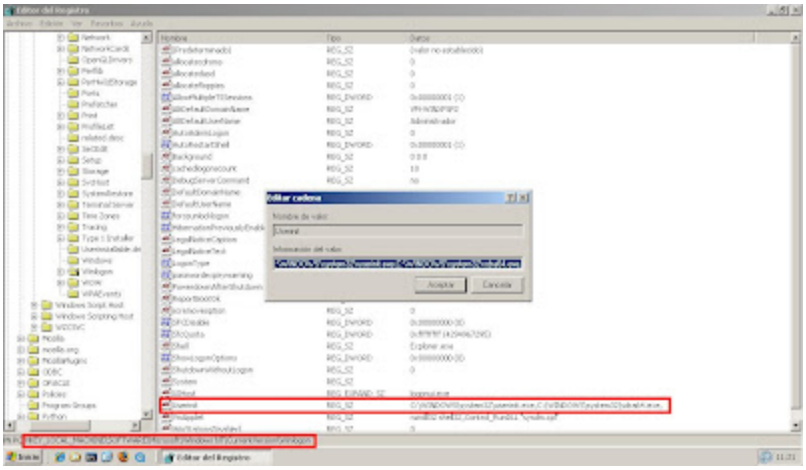
Nowadays the usual binary name is sdra64.exe and its configuration directory c:\windows\system32\lowsec, but this can change depending on the version. We already mentioned the various names for the configuration files, so now I'll only comment the different names for binary files:

> ntos.exe
> oembios.exe
> twext.exe
> twex.exe
> sdra64.exe
> bootlist32.exe
> userinit32.exe
> bootwindows.exe

- Windows registry
  Searching the Windows registry is another way to detect the infection. The trojan is able to execute itself after each reboot thanks to the inclusion of the binary path in the following registry entry:

  HKLM/SOFTWARE/Microsoft/WindowsNT/Winlogon@Userinit

  Thus simply opening the registry editor (regedit.exe) we could locate our ZeuS:



- Hooks
  ZeuS needs to put several hooks in different functions in order to make the code injection, intercept data, etc. We can find these hooks in most of the executed processes and the most common ones are the following:

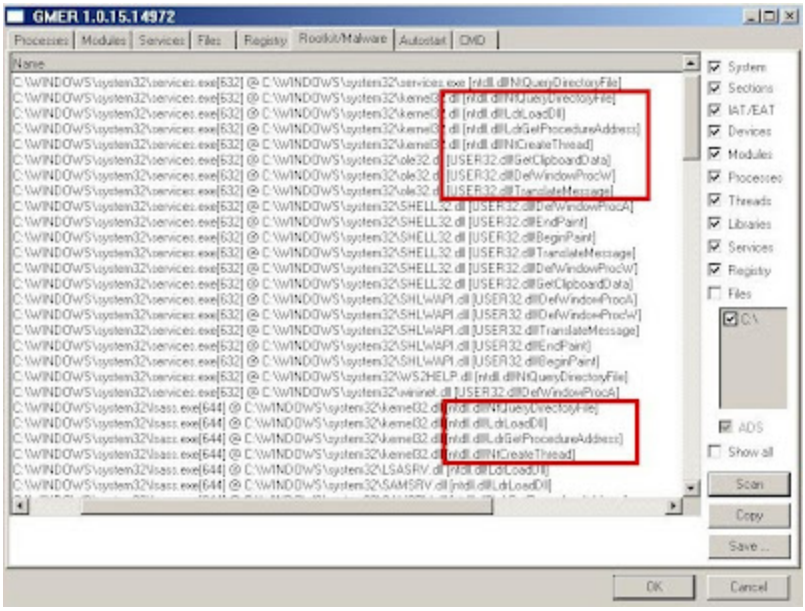
  ntdll.dll

  > NtCreateThread
  > LdrLoadDll
  > LdrGetProcedureAddress
  > NtQueryDirectoryFile

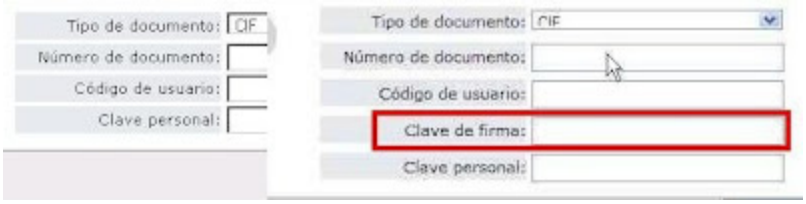
  user32.dll

  > GetClipboardData
  > TranslateMessage

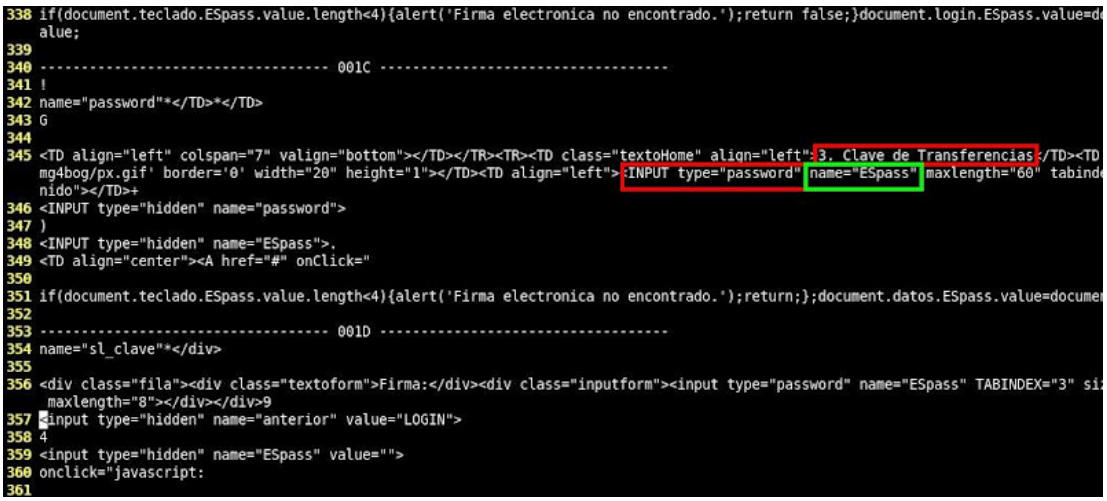
To know if we have these hooks in our system we must use an antirootkit program too:



- Online banking strange behaviour
  We can expect that people who use online banking on daily basis can notice some change in the application like an extra field asking for the password needed to make transactions or asking for all the positions of the coordinates card. This is what our trojan makes. Perhaps it will be harder to detect for people who use it occasionally, so the solution here is to pay attention while doing online banking and talk with the bank if there is any suspicious behaviour. This is an example of a login page before and after of ZeuS injection:



- Extra parameters (server-side)
  Usually when ZeuS adds extra fields in the bank page through HTML injection these additional parameters will be sent to the bank server where, depending on the injected code, could be intercepted and being possible to warn the user of a possible infection.



- Trojan mutexes
  Finally, we can detect the trojan in the system thanks to the mutexes that it creates. For example, with the OpenMutex function we can check if the ZeuS mutexes exist or not, showing the malware trace in the system. Until the moment the mutexes we have seen are:

  > __SYSTEM__64AD0625__
  > _H_64AD0625_
  > _AVIRA_2109
  > _LILO_1909
  > _SOSI_1909

Submitted by jesparza on Thu, 2009/10/01 - 12:25