INTECO Instituto Nacional de Tecnologías de la Comunicación

Merovingio

Mislead the malware

Juan Carlos Montes
Adrián Pulido

Hunting Malware like a sir
mlw.re

Index

- **Malware Analysis**
  - what else?
  - state of art
  - why?
- **PebHooking**
- **Merovingio**
  - Sandboxie
  - Merovingio Agent
- **DorianIA**
- **Merovingio Website**

Index

- **Malware Analysis**
  - what else?
  - state of art
  - why?
- **PebHooking**
- **Merovingio**
  - Sandboxie
  - Merovingio Agent
- **DorianIA**
- **Merovingio Website**

What else?

- New techniques
- Avoid signatures
- The market is dozed
- A lot of new samples every day
- It’s ~~expensive~~ complicated to have people focused on malware analysis in a CSIRT

State of art

- Commercial products are similar
  - Same VM.
  - Same drivers.
  - Same look&feel.
  - **SAME RESULTS.**
- The commercial products are the same limits
  - One sample on each VM.
    - Wait to reboot/reset the VM to start another analysis.
  - The analysis spend 2-3 minutes all times. This time is not based on the behavior of the sample.
  - Attached to the company for any grown.
  - And… the source code is not our.

Why?

- Need “anything” to detect the new samples and **behaviors**
- Avoid the dependencies of the antivirus
- Avoid the problems with VM.
  - One sample on each VM
  - Samples are out of control on execution
- Accelerate the analysis
- Include some control on the execution
- Create a system to simulate behaviors

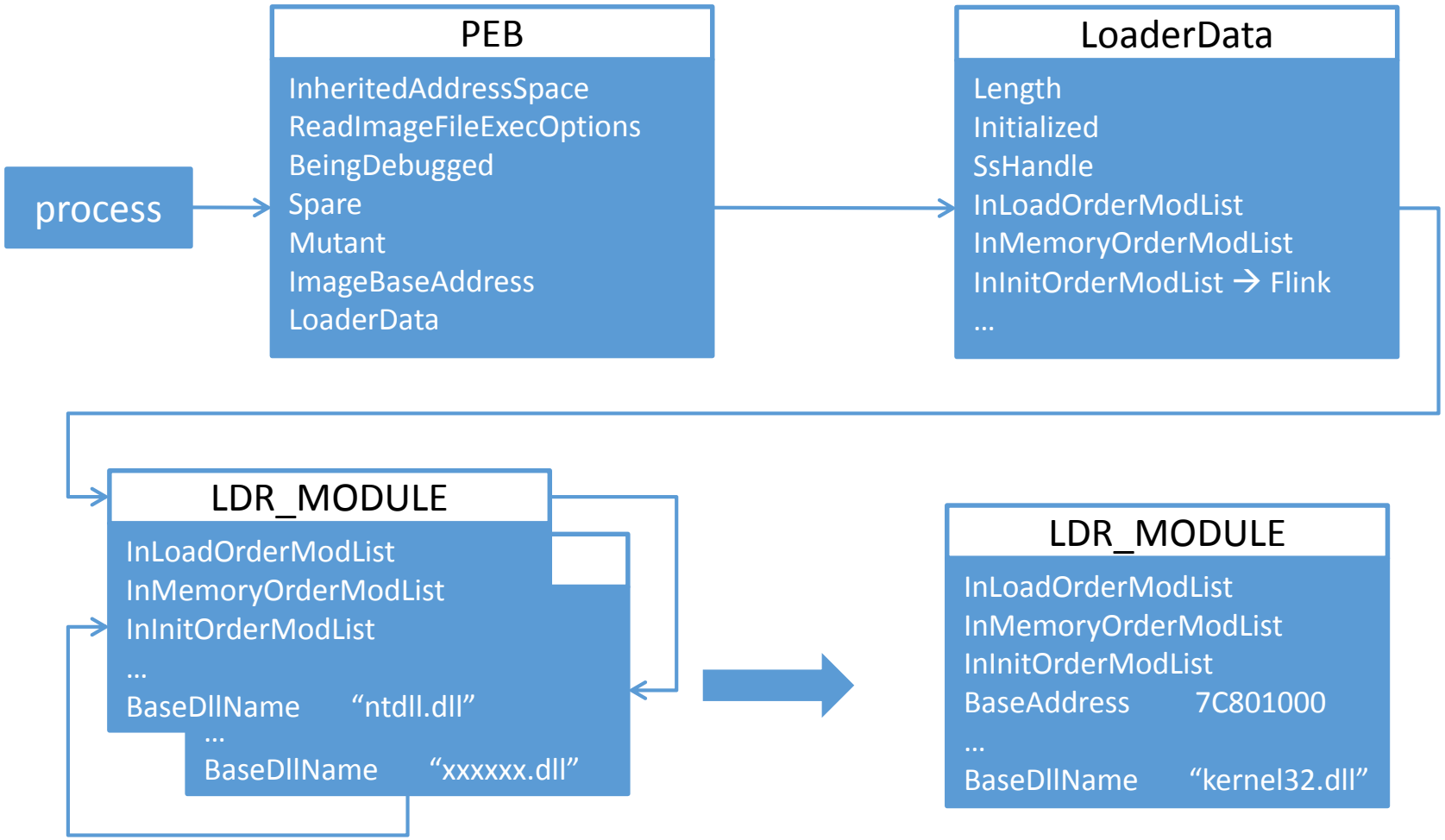
Index

- **Malware Analysis**
  - what else?
  - state of art
  - why?
- **PebHooking**
- **Merovingio**
  - Sandboxie
  - Merovingio Agent
- **DorianIA**
- **Merovingio Website**

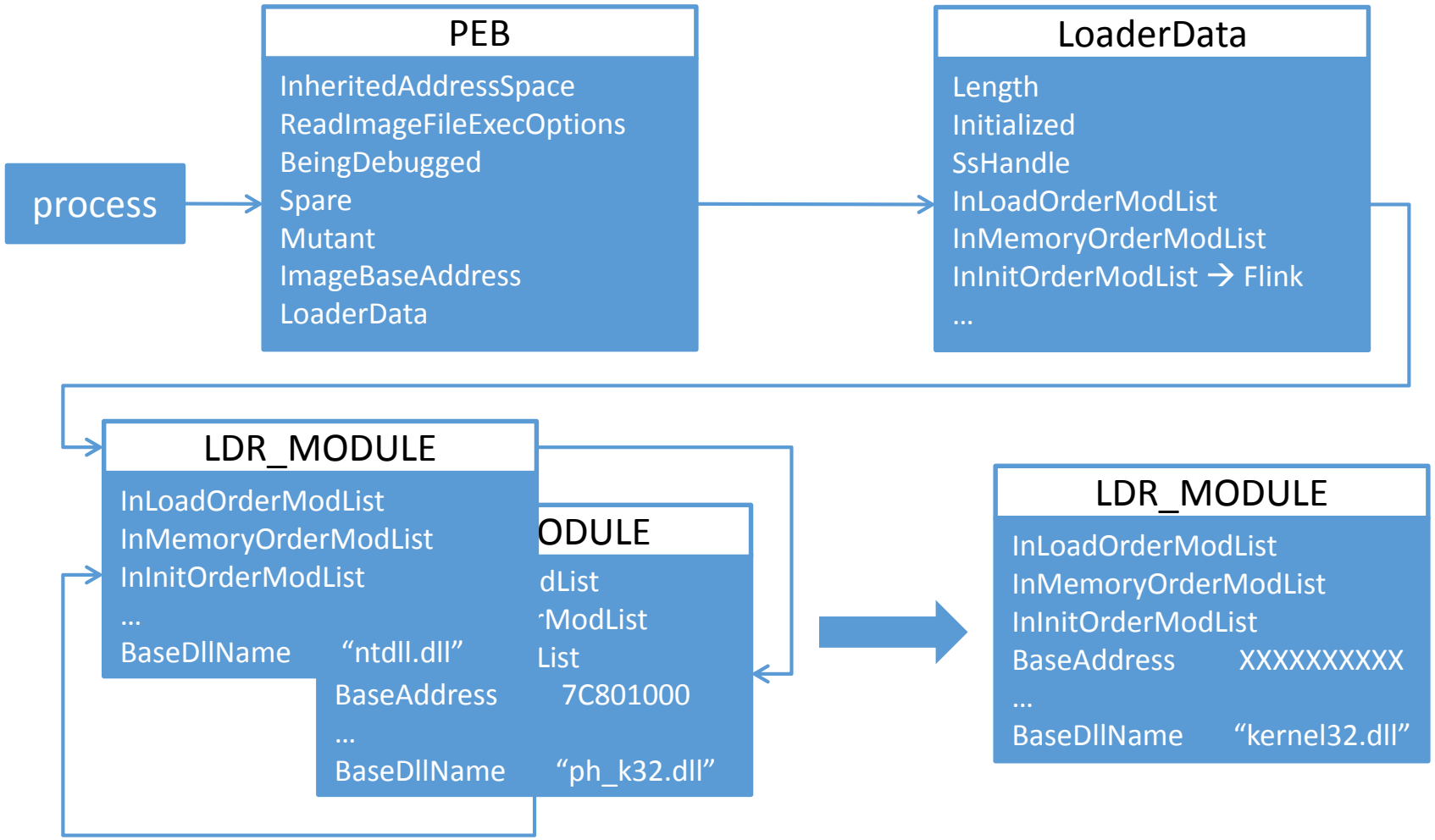
Pebhooking

- Published in Phrack #65
  - Dreg and [Shearer]
- Modify the PEB in the process to exchange real libraries for our libraries
- All dynamic loaded libraries will be hooked
- Only is necessary repair the main IAT

Pebhooking


process
PEB
InheritedAddressSpace
ReadImageFileExecOptions
BeingDebugged
Spare
Mutant
ImageBaseAddress
LoaderData
LoaderData
Length
Initialized
SsHandle
InLoadOrderModList
InMemoryOrderModList
InInitOrderModList → Flink
…
LDR_MODULE
InLoadOrderModList
InMemoryOrderModList
InInitOrderModList
…
BaseDllName “ntdll.dll”
…
BaseDllName “xxxxxx.dll”
LDR_MODULE
InLoadOrderModList
InMemoryOrderModList
InInitOrderModList
BaseAddress 7C801000
…
BaseDllName “kernel32.dll”

Pebhooking



Pebhooking

- ph_ker32.dll
  - Export the same functions that kernel32.dll
    - We must do a specific dll for each service pack
  - The functions exported have the same ordinal as the original function
  - We can manage any function we want
    - Store the return value
    - Modify params in runtime
    - Block the execution on any API

Pebhooking

process → Working…
Download binary
Execute directly from memory
or
Download binary
Write binary in disk
……
…… → My system

process → Working…
Download binary
Execute directly from memory
or
Download binary
Write binary in disk
……
…… → PEBHook → My system
Download binary
Copy binary to safe zone
or
Download binary
Copy & execute & …

Sleep,GettickCount…

Pebhooking: Source DEFS


```
CloseHandle=_CloseHandle@4                          @ 50
CompareStringA=_CompareStringA@24                   @ 56
CompareStringW=_CompareStringW@24                   @ 57
CreateFileW=_CreateFileW@28                         @ 83
CreateFileA=_CreateFileA@28                         @ 80
CreateProcessA=_CreateProcessA@40                   @ 99
CreateProcessW=_CreateProcessW@40                   @ 103
CreateRemoteThread=_CreateRemoteThread@28           @ 104
DeleteFileA=_DeleteFileA@4                          @ 130
DeleteFileW=_DeleteFileW@4                          @ 131
DeviceIoControl=_DeviceIoControl@32                 @ 137
ExitProcess=_ExitProcess@4                          @ 183
ExitThread=_ExitThread@4                            @ 184
LoadLibraryA=_LoadLibraryA@4                        @ 578
LoadLibraryW=_LoadLibraryW@4                        @ 581
MapViewOfFile=_MapViewOfFile@20                     @ 600
MapViewOfFileEx=_MapViewOfFileEx@24                 @ 601
OpenProcess=_OpenProcess@12                         @ 629
OutputDebugStringA=_OutputDebugStringA@4            @ 636
OutputDebugStringW=_OutputDebugStringW@4            @ 637
ReadFile=_ReadFile@20                               @ 676
ReadProcessMemory=_ReadProcessMemory@20             @ 679
Sleep=_Sleep@4                                      @ 831
SleepEx=_SleepEx@8                                  @ 832
WinExec=_WinExec@8                                  @ 896
WriteFile=_WriteFile@20                             @ 908
WriteProcessMemory=_WriteProcessMemory@20           @ 917
lstrcmpA=_lstrcmpA@8                                @ 936
lstrcmpW=_lstrcmpW@8                                @ 937
```

Pebhooking: Source OpenProcess


```
/************************* OpenProcess */
DLLEXPORT
HANDLE _stdcall _OpenProcess( DWORD dwDesiredAccess, BOOL bInheritHandle, DWORD dwProcessId )
{
    char log[MAX_PATH*2];
    char * pname;
    HANDLE hProcess = OpenProcess( dwDesiredAccess, bInheritHandle, dwProcessId );

    pname = showProcessInformation( dwProcessId );
    if (pname != 0)
    {
        sprintf(log, "0x%x|%s|0x%x", hProcess, pname, dwProcessId );
        free(pname);
    }
    ApiLogger( "OpenProcess", log );

    return hProcess;
}
```

Pebhooking: Source ReadProcessMemory

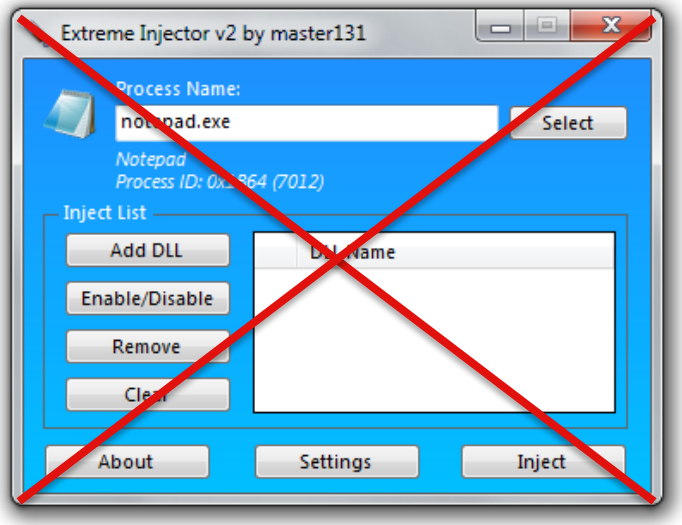
```
/************************* ReadProcessMemory */
DLLEXPORT
BOOL _stdcall _ReadProcessMemory( HANDLE hProcess, LPCVOID lpBaseAddress,
     LPVOID lpBuffer, SIZE_T nSize, SIZE_T *lpNumberOfBytesRead )
{
    char * b64str;
    char * logentry;
    size_t b64_size;
    BOOL ret = ReadProcessMemory( hProcess, lpBaseAddress, lpBuffer,
          nSize, lpNumberOfBytesRead );

    // Alloc buffers
    logentry = VirtualAlloc(NULL, nSize*3, MEM_COMMIT, PAGE_READWRITE);

    // Base64
    build_decoding_table();
    b64str = base64_encode( lpBuffer, nSize, &b64_size );
    b64str[b64_size] = '\x00';
    base64_cleanup();

    // Log the API
    sprintf(
        logentry,
        "0x%x|0x%x|0x%x|%s",
        hProcess,
        lpBaseAddress,
        nSize,
        b64str
    );
    ApiLogger( "ReadProcessMemory", logentry );

    // Free mem
    free(b64str);
    VirtualFree(logentry, 0, MEM_RELEASE);

    return ret;
```

Index

- **Malware Analysis**
  - what else?
  - state of art
  - why?
- **PebHooking**
- **Merovingio**
  - Origin
  - Sandboxie
  - Merovingio Agent
- **DorianIA**
- **Merovingio Website**

Origin

- Need to inject DLL in all processes
- Retrieves logs and info
- Avoid infecting the machine
- Multithread
- Does a lot of analysis
- Use(r) friendly
- Controls the previous analysis
- …



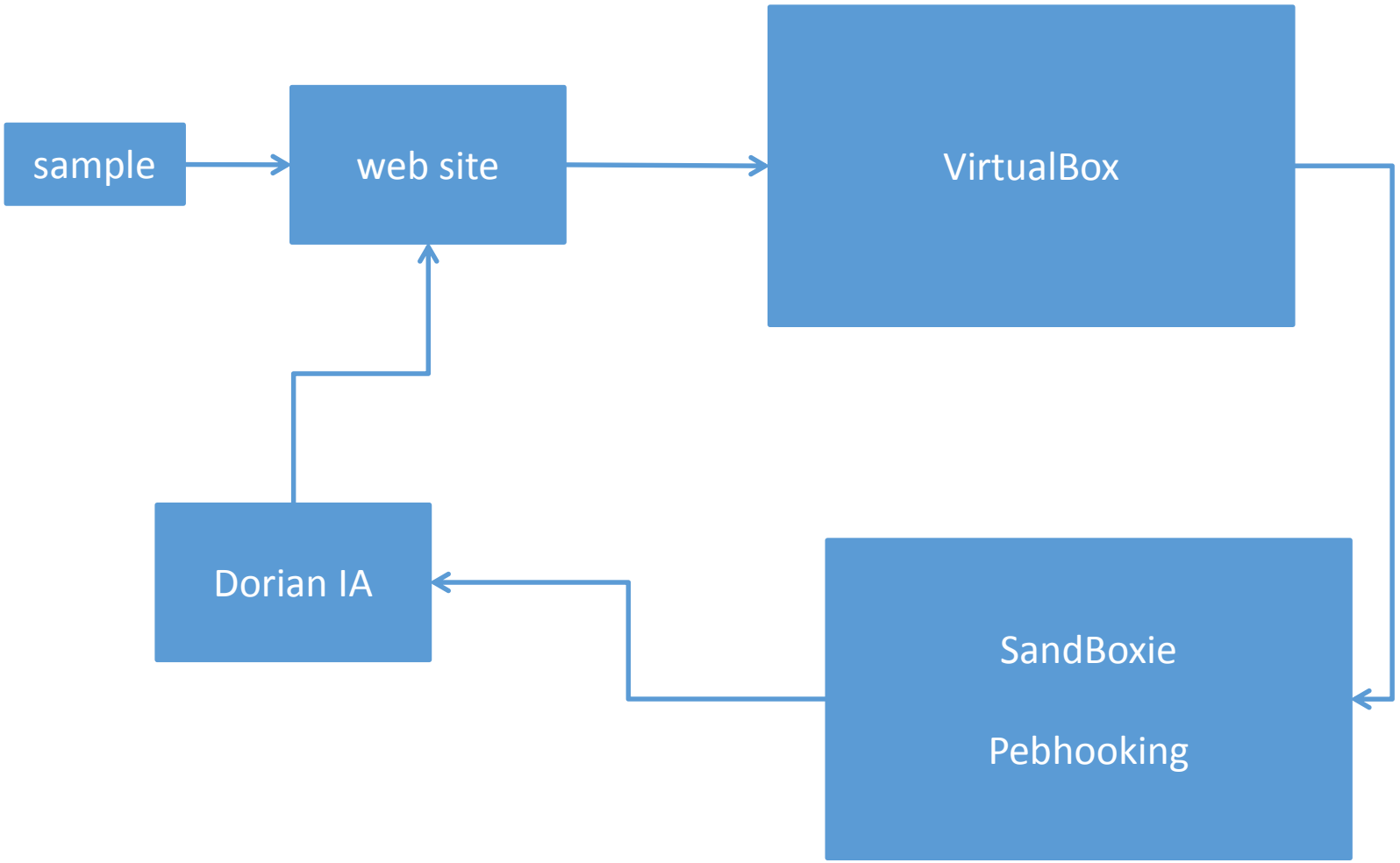
Contiene: 637 archivos, 194 carpetas

Merovingio

- “Virtual Machine”
- Sandboxie
- Pebhooking
- DorianIA

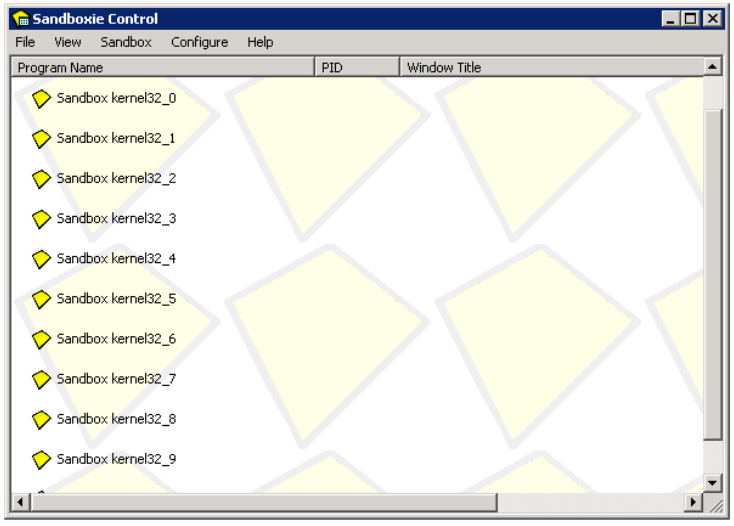
- And… this is the web! ☺

Merovingio


sample
web site
VirtualBox
Dorian IA
SandBoxie
Pebhooking

Sandboxie

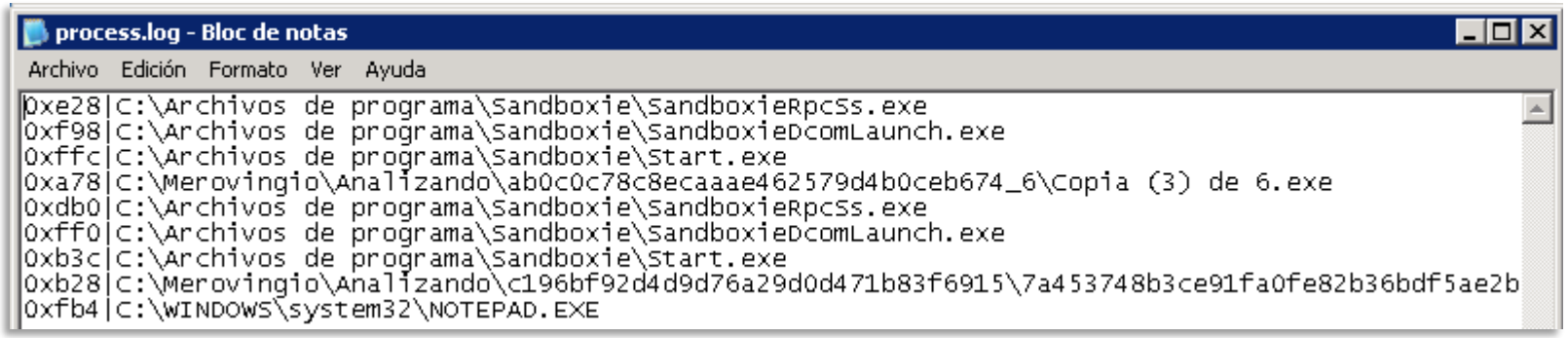
- Runs programs in a sandbox
- Prevents permament changes on the system
- Helps us to load our libraries in each process
- Isolates each program execution

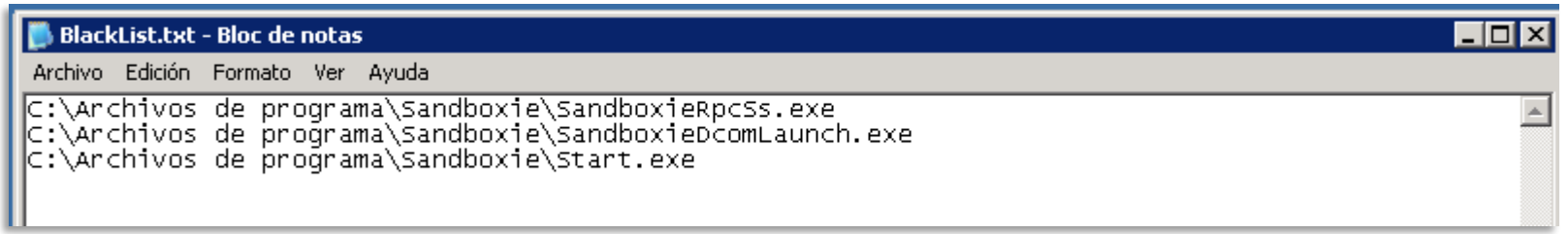



```
[kernel32_0]

InjectDll=C:\proyectos\phook\bin\windows_XP_SP2\ph_ker32.dll
InjectDll=C:\windows\system32\user32.dll
InjectDll=C:\windows\system32\advapi32.dll
InjectDll=C:\windows\system32\SHELL32.dll
InjectDll=C:\windows\system32\imagehlp.dll
```

SandBoxie

INTECO Instituto Nacional de Tecnologías de la Comunicación





Merovingio Agent

- Tested in Windows XP and Windows 7
- Developed in Python v2.7
- Can manage as many sandboxed instances as we want
- Recover the logs and send us to next step
- Multithread
- Can receive more than one sample at the same time
- Decide on which instance must be executed the sample
  - Free slot
  - Specific analysis
- Monitorizes the analysis to detected when it is finishing

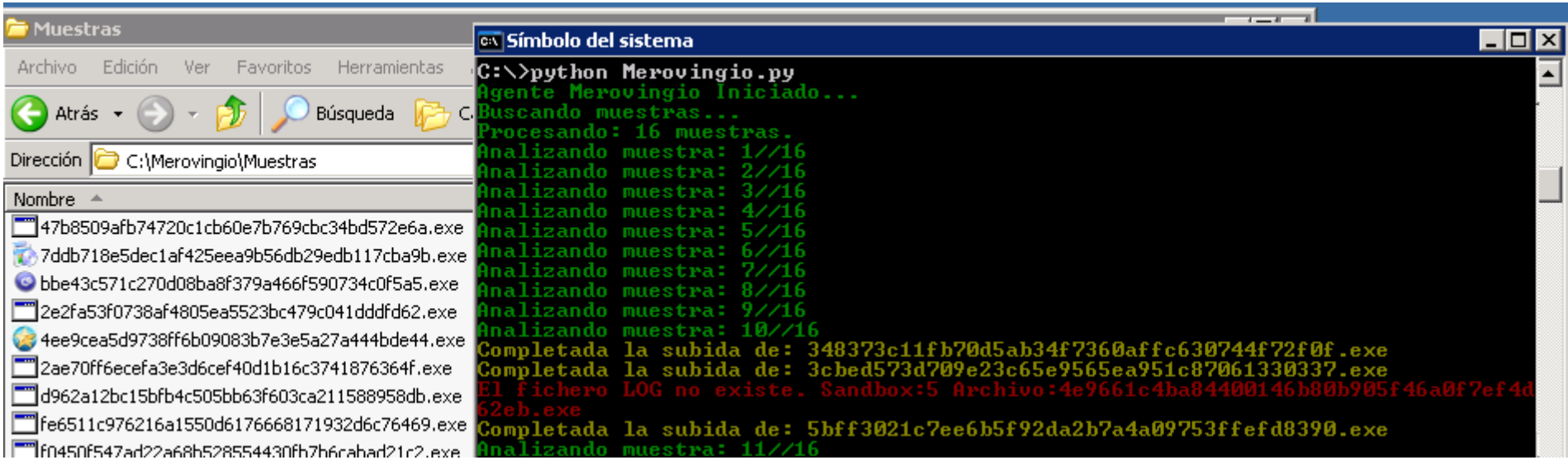
Merovingio Agent

INTECO Instituto Nacional de Tecnologías de la Comunicación

```
import threading
import sys
import os
import shutil
import hashlib
import time
import ftplib
from colorama import Fore,Back,Style
from array import *

RutaAnalisis = "C:\\Merovingio\\Analizando\\"
RutaMuestras = "C:\\Merovingio\\Muestras\\"
RutaMerovingio = "C:\\Merovingio\\"
RutaSandbox = "C:\\Sandbox\\tecnico\\"
NumSandbox = 40 #Ampliar en caso de querer utilizar más. Debe ampliarse en sandboxie
SandboxEnUso = array('i',[])
FTPServidor = '10.xx.xx.xx'
FTPUsuario = 'merovingio'
FTPClave = '                                    '
TiempoAnalizando = 60 #Establecido en segundos

def main() :
    MensajeAplicacion ("Agente Merovingio Iniciado...")
    files = os.listdir(RutaAnalisis) #Limpia la carpeta de analisis. Por si hubo errores
    for file in files:
        shutil.rmtree (RutaAnalisis+file ,True)
    for i in range(NumSandbox):
        SandboxEnUso.append(int(i))
        SandboxEnUso[i] = 0
    HiloLimpiaSandbox = threading.Thread( target=Liberar_Sandbox, name='limpia sandbox')
    HiloLimpiaSandbox.start()
    MensajeAplicacion ("Buscando muestras...")
    while 1:
        Control_Muestras()
        time.sleep(5)
    return 0
```

Merovingio Agent

- Searches new samples on the path
- Copies the new samples and analyzes them
- Monitorizes Sandboxie's box
- Retrieve logs on the website



Merovingio Agent


```
int main( void )
{
    HANDLE hFile;
    HMODULE hModule;
    HKEY hKey;
    LONG lResult;
    DWORD dwValue, dwType, dwSize = sizeof(dwValue);
    LPTSTR lpValueName = "Install_Dir";
    DWORD dwDefault = 0x00000000;
    char data[100] = "TEST1";
    char out1[100] = "";
    char out2[100] = "";
    int b64_size = 0;

    // CreateFile
    hFile = CreateFileW( L"file1", GENERIC_WRITE, 0, NULL, CREATE_ALWAYS, 0, NULL );
    if ( hFile != INVALID_HANDLE_VALUE )
        CloseHandle( hFile );

    hFile = CreateFileA( "file2", GENERIC_WRITE, 0, NULL, CREATE_ALWAYS, 0, NULL );
    if ( hFile != INVALID_HANDLE_VALUE )
        CloseHandle( hFile );

    hFile = CreateFileW( L"file3", GENERIC_WRITE, 0, NULL, CREATE_ALWAYS, 0, NULL );
    if ( hFile != INVALID_HANDLE_VALUE )
        CloseHandle( hFile );

    // DeleteFile
    //DeleteFileA( "C:\\file3" );
    //DeleteFileW( L"C:\\file2" );
    DeleteFileA( "file1" );
```

NULL

Merovingio Agent

INTECO Instituto Nacional de Tecnologías de la Comunicación

DEMO

Index

- **Malware Analysis**
  - what else?
  - state of art
  - why?
- **PebHooking**
- **Merovingio**
  - Sandboxie
  - Merovingio Agent
- **DorianIA**
- **Merovingio Website**

Dorian IA

- It is based on the workflows of the neural networks
- Set the time in each received log
- Analyze the log looking for patterns
- Create execution blocks
- Try to link the different blocks to create behaviors
- Show the results in a new log that is send to the website
- At the moment it can learn new behaviors, our aim is to create a real AI

Dorian IA

Log from PebHooking
LoadLibraryW|IMM32.DLL
CreateFileW | C:\ikkka.exe|0x178
CreateFileW|COMCTL32.DLL|0x4C
LoadLibraryW|user32.dll
WriteFile | 0x178 | 0x22800 | XXXXXXXXX
CloseHandle | 0x4C
CloseHandle | 0x178

Block
CreateFileW | C:\ikkka.exe | 0x178
WriteFile | 0x178 | 0x22800 | XXXXXXXXXX
CloseHandle | 0x178

Dorian IA

Log from PebHooking
LoadLibraryW|IMM32.DLL
CreateFileW | C:\itself.exe|0x77
ReadFile | 0x22800 | XXXXXXXXX
CloseHandle | 0x77
DeleteFile | C:\autoexec.bat
CreateFileW | C:\ikkka.exe|0x178
CreateFileW|COMCTL32.DLL|0x4C
LoadLibraryW|user32.dll
WriteFile | 0x178 | 0x22800 | XXXXXXXXX
CloseHandle | 0x4C
CloseHandle | 0x178

Block
CreateFileW | C:\itself.exe|0x77
ReadFile | 0x22800 | XXXXXXXXX
CloseHandle | 0x77

Block
CreateFileW | C:\ikkka.exe | 0x178
WriteFile | 0x178 | 0x22800 | XXXXXXXXXX
CloseHandle | 0x178

Block
DeleteFile | C:\autoexec.bat

- Read itself
- Write itself in other file
- Similar content: we use ssdeep to compare the information with threshold 95%

The sample was copied itself to another path.

Dorian IA: Source Rules

Rules:
-Self replicate

-Duplicate others files

-Delete it self

-Open process

- …

```
<sidsearch list="selfreplicate" param="param1" compareparam="process.
  <desc>SELF-REPLICATE</desc>
  <showvalue values="3">[{}%]: "{}" => "{}"</showvalue>
  <malware>200</malware>
  <api>CreateFileA</api>
  <api>CreateFileW</api>
  <proc>ReadFile</proc>
  <related>WriteFile</related>
</sidsearch>
<!-- Locate replications -->
<sidsearch list="duplicatedfiles" param="param1" ssdeepcompare="paran
  <desc>DUPLICATED FILES</desc>
  <showvalue values="3">[{}%]: {} => {}</showvalue>
  <malware>75</malware>
  <api>CreateFileA</api>
  <api>CreateFileW</api>
  <proc>ReadFile</proc>
  <related>WriteFile</related>
</sidsearch>
<!-- Locate self-delete -->
<sidsearch list="deleteitself" param="param1" funcparam="param1">
  <desc>DELETE ITSELF</desc>
  <malware>199</malware>
  <api>DeleteFileA</api>
  <api>DeleteFileW</api>
  <func>getProcessName</func>
</sidsearch>
<!-- Locate multiple OpenProcess -->
<sidsearch check="openprocess" times="5">
  <desc>MULTIPLE OPENPROCESS CALLED</desc>
  <malware>75</malware>
  <api>OpenProcess</api>
</sidsearch>
<!-- Locate remote threads -->
<sidsearch check="remotethread">
```

Index

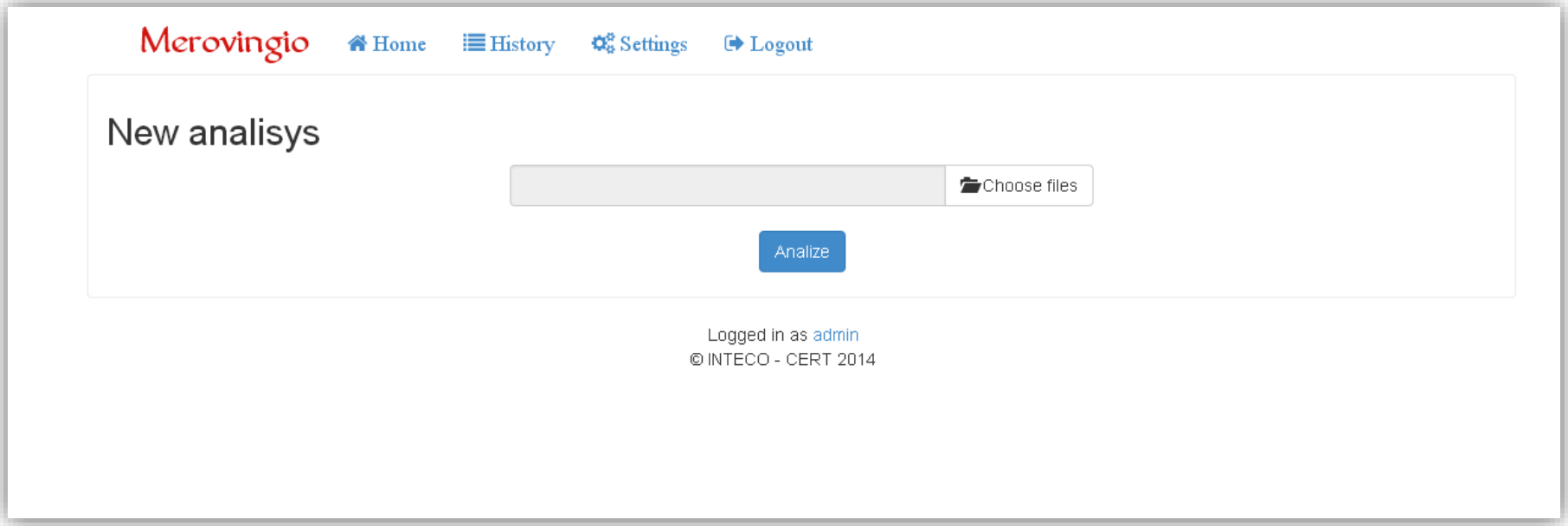
- **Malware Analysis**
  - what else?
  - state of art
  - why?
- **PebHooking**
- **Merovingio**
  - Sandboxie
  - Merovingio Agent
- **DorianIA**
- **Merovingio Website**

Website Features

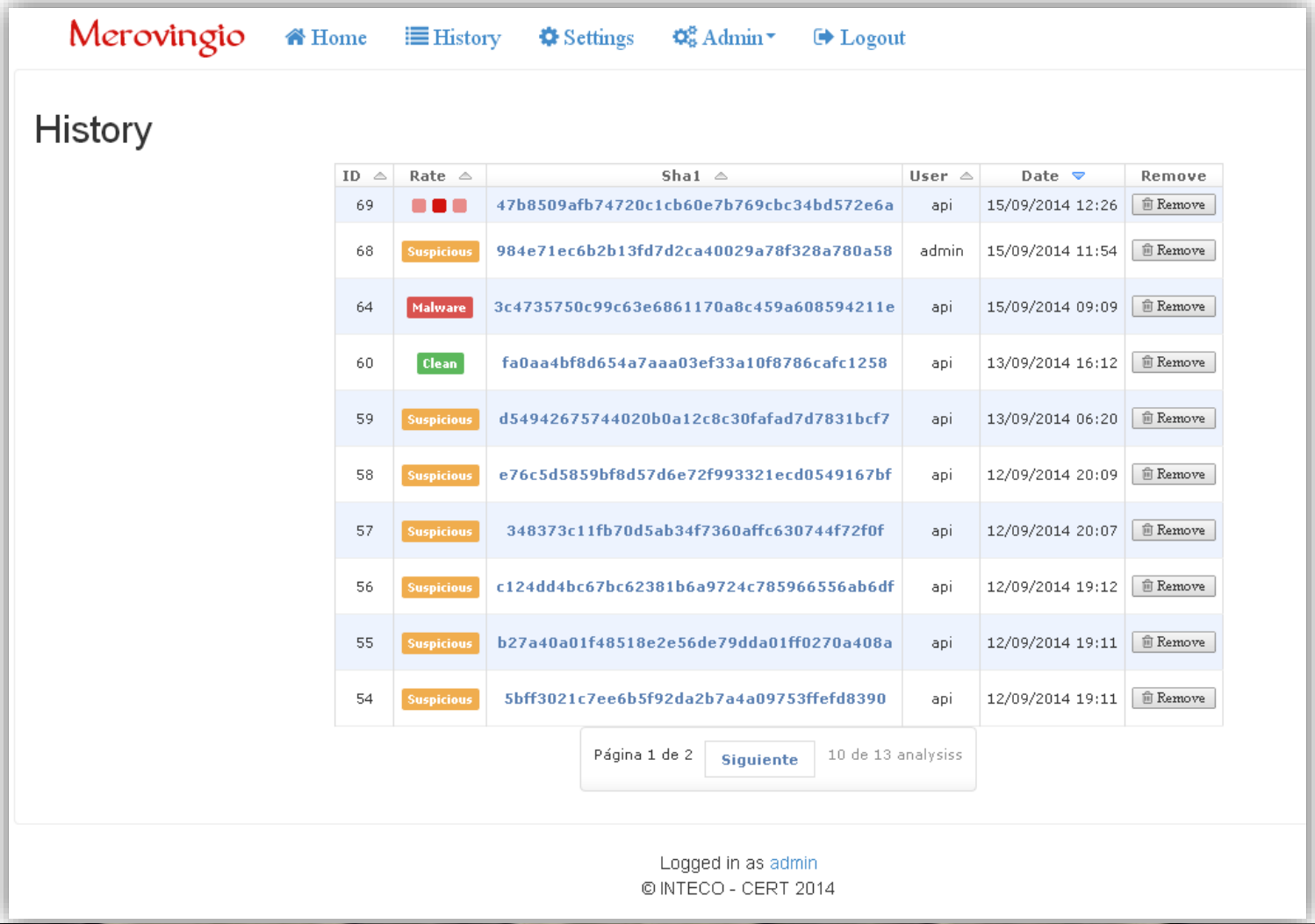
- User management
- Able to upload different samples at the same time
- Hold the history to recover old reports
- Look for the samples by its filename or hash (SHA256,SHA1,MD5)
- All the communication with the agent is transparent to user
- Find malicious samples easily from the history

INTECO Instituto Nacional de Tecnologías de la Comunicación

Home page / Send samples



Website Screenshots



Website Screenshots

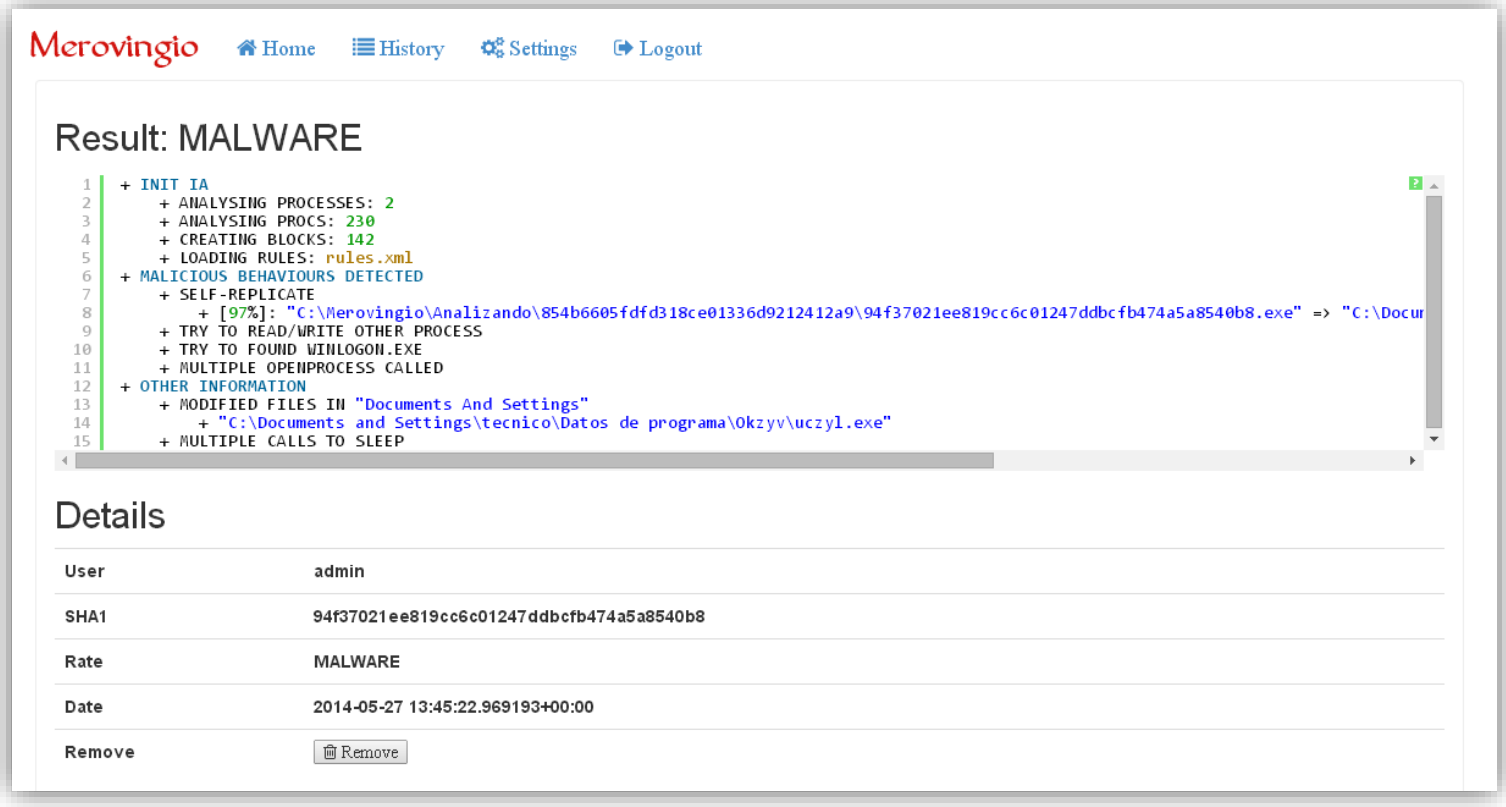
INTECO Instituto Nacional de Tecnologías de la Comunicación

Raw log

```
2014-03-10 14:46:30.35600|0xf4|zeus.exe|Sleep|20
2014-03-10 14:46:30.36700|0xf4|zeus.exe|Sleep|20
2014-03-10 14:46:30.37700|0xf4|zeus.exe|Sleep|20
2014-03-10 14:46:30.38700|0xf4|zeus.exe|Sleep|20
2014-03-10 14:46:30.39700|0xf4|zeus.exe|Sleep|20
2014-03-10 14:46:30.40700|0xf4|zeus.exe|Sleep|20
2014-03-10 14:46:30.41700|0xf4|zeus.exe|CreateFileW|C:\zeus.exe|0x174
2014-03-10 14:46:30.41700|0xf4|zeus.exe|ReadFile|0x174|0x22800|TVoAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
2014-03-10 14:46:30.43700|0xf4|zeus.exe|CloseHandle|0x174
2014-03-10 14:46:30.52700|0xf4|zeus.exe|CreateFileW|C:\Documents and Settings\Tecnico\Datos de programa\Tysow\ovcu.exe|0x174
2014-03-10 14:46:30.52700|0xf4|zeus.exe|WriteFile|0x174|0x22800|TVoAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
2014-03-10 14:46:30.52700|0xf4|zeus.exe|CloseHandle|0x174
2014-03-10 14:46:30.52700|0xf4|zeus.exe|CreateFileW|C:\Documents and Settings\Tecnico\Datos de programa|0x174
2014-03-10 14:46:30.52700|0xf4|zeus.exe|CloseHandle|0x174
2014-03-10 14:46:30.53700|0xf4|zeus.exe|CreateFileW|C:\Documents and Settings\Tecnico\Datos de programa\Tysow\ovcu.exe|0x174
2014-03-10 14:46:30.53700|0xf4|zeus.exe|CloseHandle|0x174
2014-03-10 14:46:30.53700|0xf4|zeus.exe|CreateFileW|C:\Documents and Settings\Tecnico\Datos de programa\Tysow|0xffffffff
2014-03-10 14:46:30.53700|0xf4|zeus.exe|CreateFileW|C:\Documents and Settings\Tecnico\Datos de programa\Teylq\episw.pua|0x174
2014-03-10 14:46:30.53700|0xf4|zeus.exe|CloseHandle|0x174
2014-03-10 14:46:30.53700|0xf4|zeus.exe|CreateFileW|C:\Documents and Settings\Tecnico\Datos de programa\Teylq|0xffffffff
2014-03-10 14:46:30.53700|0xf4|zeus.exe|CreateFileW|C:\Documents and Settings\Tecnico\Datos de programa|0x174
2014-03-10 14:46:30.54700|0xf4|zeus.exe|CloseHandle|0x17c
2014-03-10 14:46:30.55700|0xf4|zeus.exe|MapViewOfFile|0x180|0x22800
2014-03-10 14:46:30.55700|0xf4|zeus.exe|CloseHandle|0x180
2014-03-10 14:46:30.58700|0xf4|zeus.exe|CloseHandle|0x188
2014-03-10 14:46:30.58700|0xf4|zeus.exe|MapViewOfFile|0x18c|0x22800
2014-03-10 14:46:30.58700|0xf4|zeus.exe|CloseHandle|0x18c
```

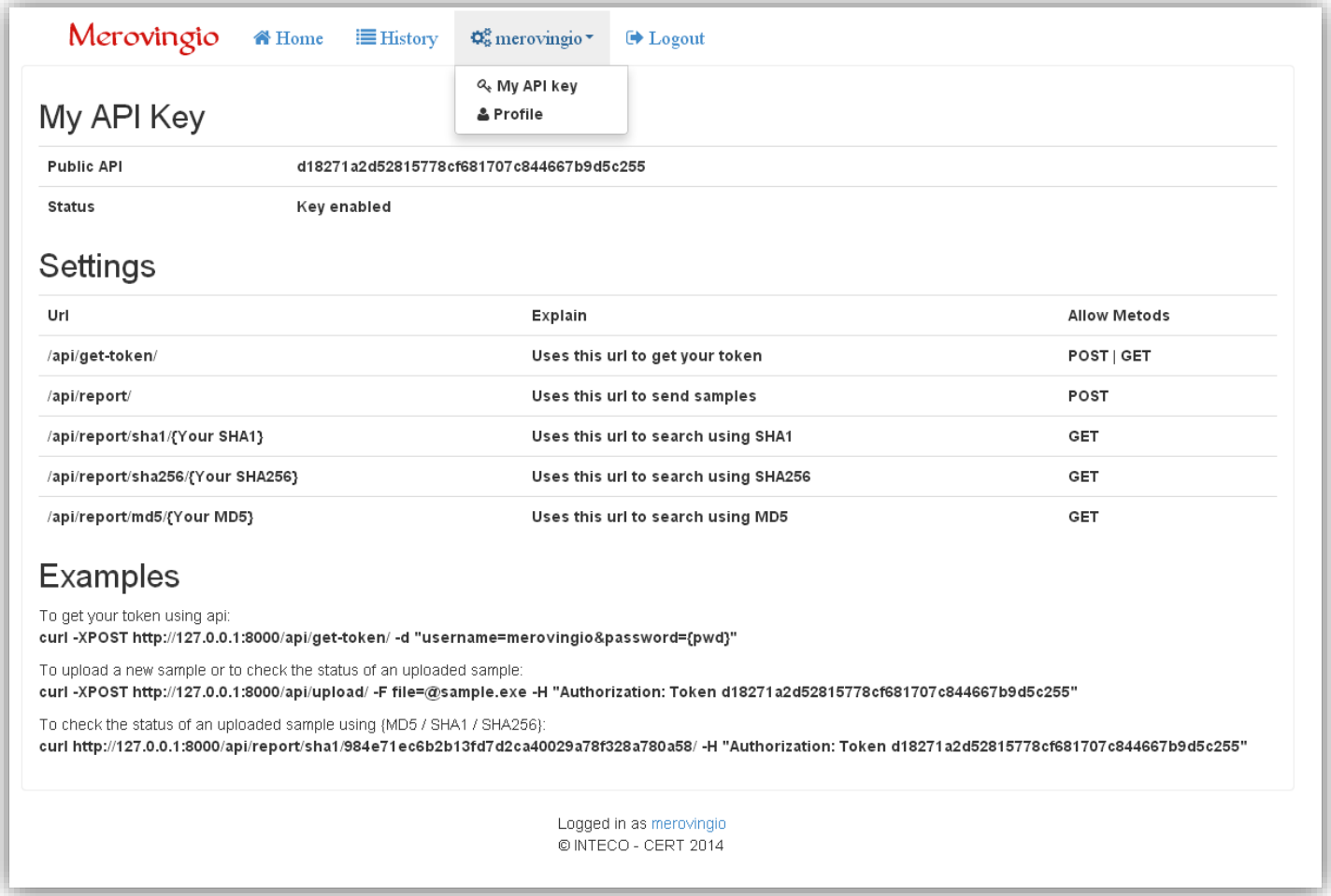
Website Screenshots

Analysis



Website Screenshots

API



Website Achivements

- Max. runtime 2 minutes, but the analysis stop when it doesn't detect any new behavior
- The same machine can analyze over 20 samples both (VM or real)
- To grow we need add RAM memory to allocate more process or add a new machine to get 20 slots.
- Very cheap (information for 20 analysis):
  - Just one machine
  - 4Ghz CPU (4 cores) and 4Gb RAM
  - The analysis can be stopped when the sample finishes the execution

Merovingio Numbers

- **720** samples can be analyzed in each sandboxed instance daily
- **14.400** samples use 20 instances in the sandboxie
- Only 1 cheap machine to get this numbers

Questions??

- Any doubs??
- Any questions??
- Any donations?? ☺
- Donaciones?? ☺ ☺ ☺

Gracias!!!

Juan C. Montes

Mail personal: **jcmontes.tec@gmail.com**

Mail CERT: **jcmontes@cert.inteco.es**

Twitter: **@jcmontes_tec**

Adrián Pulido

Mail personal: **winsock@gmail.com**

Mail CERT: **adrian.pulido@inteco.es**

Twitter: **@winsock**