Exploring the complexities of Digital Forensics

BY: Mohamed Ali Aoun

1. Introduction: Importance of Digital Forensics

In today's digital era, where information is stored in cyberspace, digital forensics emerges as a vital tool in uncovering the truth. With its intricate methodologies and advanced techniques, it delves deep into the digital realm, analyzing data trails, unraveling hidden connections, and exposing cybercrime. Join us on an enlightening journey as we explore the intricacies of digital forensics, revealing the undeniable truth in the digital age.

2. Definition and Scope of Digital Forensics

Digital forensics, also known as computer forensics, is the branch of forensic science that involves the collection, preservation, analysis, and presentation of electronic evidence in legal investigations. It encompasses various areas, including network forensics, mobile device forensics, and forensic data analysis. By employing specialized tools and techniques, digital forensics professionals help uncover crucial evidence and provide insights into cybercrimes, frauds, and other digital malfeasances.

3. guiding principles and proven techniques in Digital Forensics

Preservation: Ensuring the pristine condition of evidence is paramount, minimizing the risk of alteration or contamination.

Documentation: Meticulously recording every step taken, from initial seizure to analysis and reporting, is crucial for transparency and legal defensibility.

Chain of Custody: Maintaining a clear and unbroken record of who handled the evidence and when safeguards its integrity and admissibility in court.

Data Acquisition: Capturing digital evidence from various sources, like hard drives, mobile devices, and cloud storage, forms the initial step.

Imaging: Creating an exact replica of a storage device is critical for preserving its entire contents, including deleted or hidden data.

Hashing: Generating unique mathematical fingerprints of digital evidence ensures its authenticity and verifies its unaltered state throughout the investigation.

Analysis: Employing specialized software to examine and interpret the acquired data, uncovering hidden messages, file traces, and other digital artifacts.

Context : This is a challenge that I successfully solved during a CTF competition called Hackfest

4.Real life challenge

this is a .raw memory dump file so we will be analyzing it using volatility

```
PS C:\Users\dalia\Tools\volatility_2.5.win.standalone\volatility_2.5.win.standalone> .\volatility-2.5.standalone.exe -f .\micro.raw imageinfo
Volatility Foundation Volatility Framework 2.5
INFO    : volatility.debug    : Determining profile based on KDBG search...
          Suggested Profile(s) : Win2008R2SP0x64, Win7SP1x64, Win7SP0x64, Win2008R2SP1x64
                     AS Layer1 : AMD64PagedMemory (Kernel AS)
                     AS Layer2 : FileAddressSpace (C:\Users\dalia\Tools\volatility_2.5.win.standalone\volatility_2.5.win.standalone\micro.raw)
                      PAE type : No PAE
                           DTB : 0x187000L
                          KDBG : 0xf800027f40a0L
          Number of Processors : 1
     Image Type (Service Pack) : 1
                KPCR for CPU 0 : 0xfffff800027f5d00L
             KUSER_SHARED_DATA : 0xfffff78000000000L
           Image date and time : 2023-06-23 18:31:21 UTC+0000
     Image local date and time : 2023-06-23 19:31:21 +0100
PS C:\Users\dalia\Tools\volatility_2.5.win.standalone\volatility_2.5.win.standalone>
```

4.Real life challnge

The profile we will be using is Win7SP1x64
We will first be proceeding with a filescan and see what the desktop holds for us:
We have two files that seemed to be interesting

```
0x000000007d949af0      2      1 R--rwd \Device\HarddiskVolume1\Users\Public\Desktop
0x000000007d94b420      2      0 RW---- \Device\HarddiskVolume1\Users\vboxuser\Desktop\confidential.rar
0x000000007d94b5f0      2      1 R--rwd \Device\HarddiskVolume1\Users\vboxuser\Desktop
0x000000007d94bb70      2      0 RW---- \Device\HarddiskVolume1\Users\vboxuser\Desktop\doc.docm
```

4.Real life challenge

Let's first download the two files and see what we have:

```
\volatility_2.5.win.standalone> .\volatility-2.5.standalone.exe -f .\micro.raw --profile=Win7SP1x64 dumpfiles -Q 0x000000007d94bb70 --d

Volatility Foundation Volatility Framework 2.5
DataSectionObject 0x7d94bb70   None   \Device\HarddiskVolume1\Users\vboxuser\Desktop\doc.docm


Volatility Foundation Volatility Framework 2.5
DataSectionObject 0x7d94b420   None   \Device\HarddiskVolume1\Users\vboxuser\Desktop\confidential.rar
```

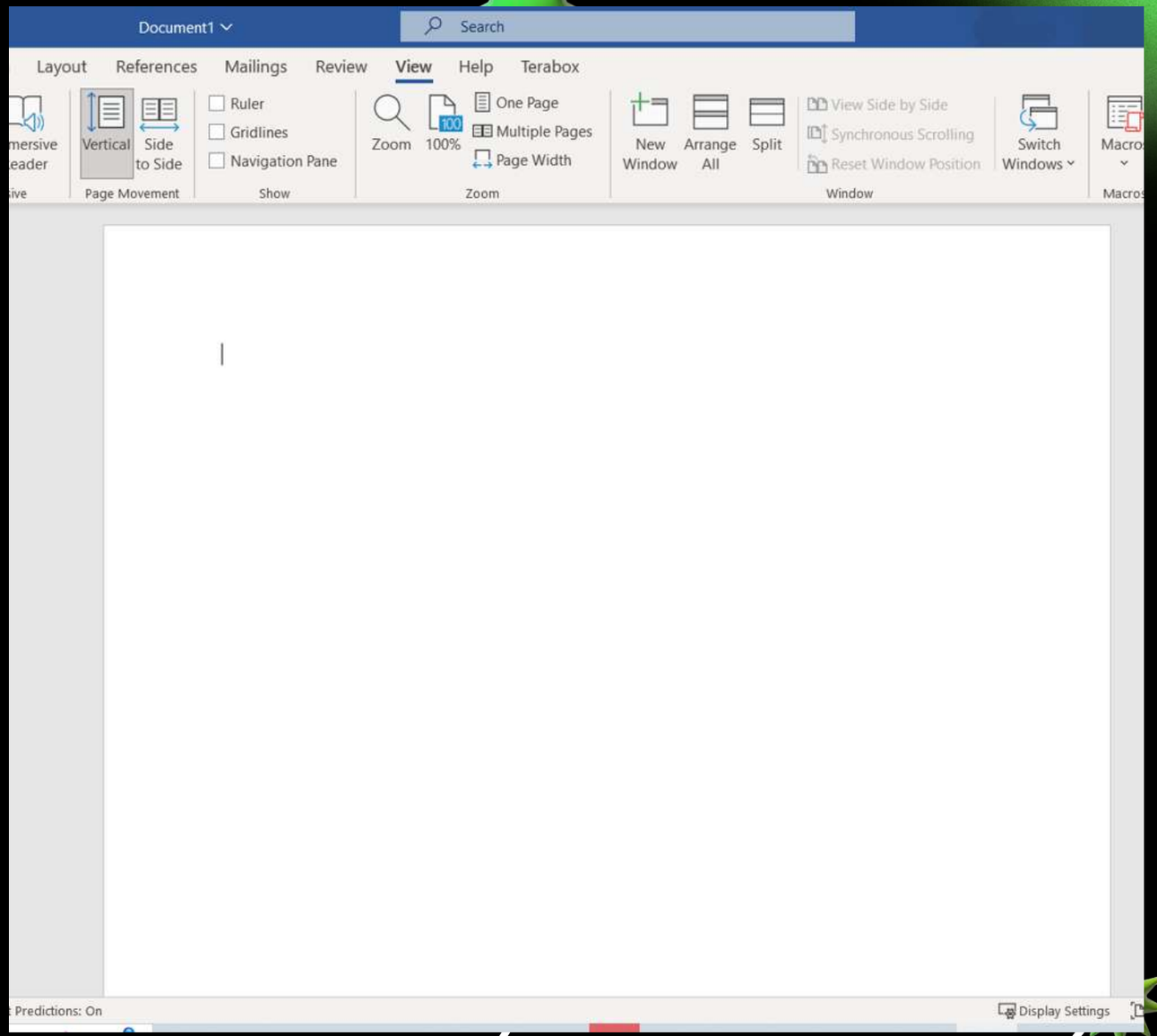
4.Real life challenge

We first unrar the confidential.rar file and we get this confidential.txt:

```
PS C:\Users\dalia\Tools\volatility_2.5.win.standalone\volatility_2.5.win.standalone> cd .\confidential\
PS C:\Users\dalia\Tools\volatility_2.5.win.standalone\volatility_2.5.win.standalone\confidential> ls


    Directory: C:\Users\dalia\Tools\volatility_2.5.win.standalone\volatility_2.5.win.standalone\confidential


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         6/23/2023   7:18 PM                confidential


PS C:\Users\dalia\Tools\volatility_2.5.win.standalone\volatility_2.5.win.standalone\confidential> cd .\confidential\
PS C:\Users\dalia\Tools\volatility_2.5.win.standalone\volatility_2.5.win.standalone\confidential\confidential> ls


    Directory: C:\Users\dalia\Tools\volatility_2.5.win.standalone\volatility_2.5.win.standalone\confidential\confidential


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         6/23/2023   7:18 PM            302 confidential.txt


PS C:\Users\dalia\Tools\volatility_2.5.win.standalone\volatility_2.5.win.standalone\confidential\confidential> cat .\confidential.txt
\x15u\xc6\xbd\x02k\xa9\xb9\xdaJV\x80\xa2\xf0'\x80!G\xc1u\x82{\xc2\xaa\xe6!\x19&\xc0\xea\x1a\x06\xc1\x9aYj\xc3\xd1\x9b\xa3\x0e\xeb\xb5\xe9\x11F\xb5\x90
PS C:\Users\dalia\Tools\volatility_2.5.win.standalone\volatility_2.5.win.standalone\confidential\confidential>
```

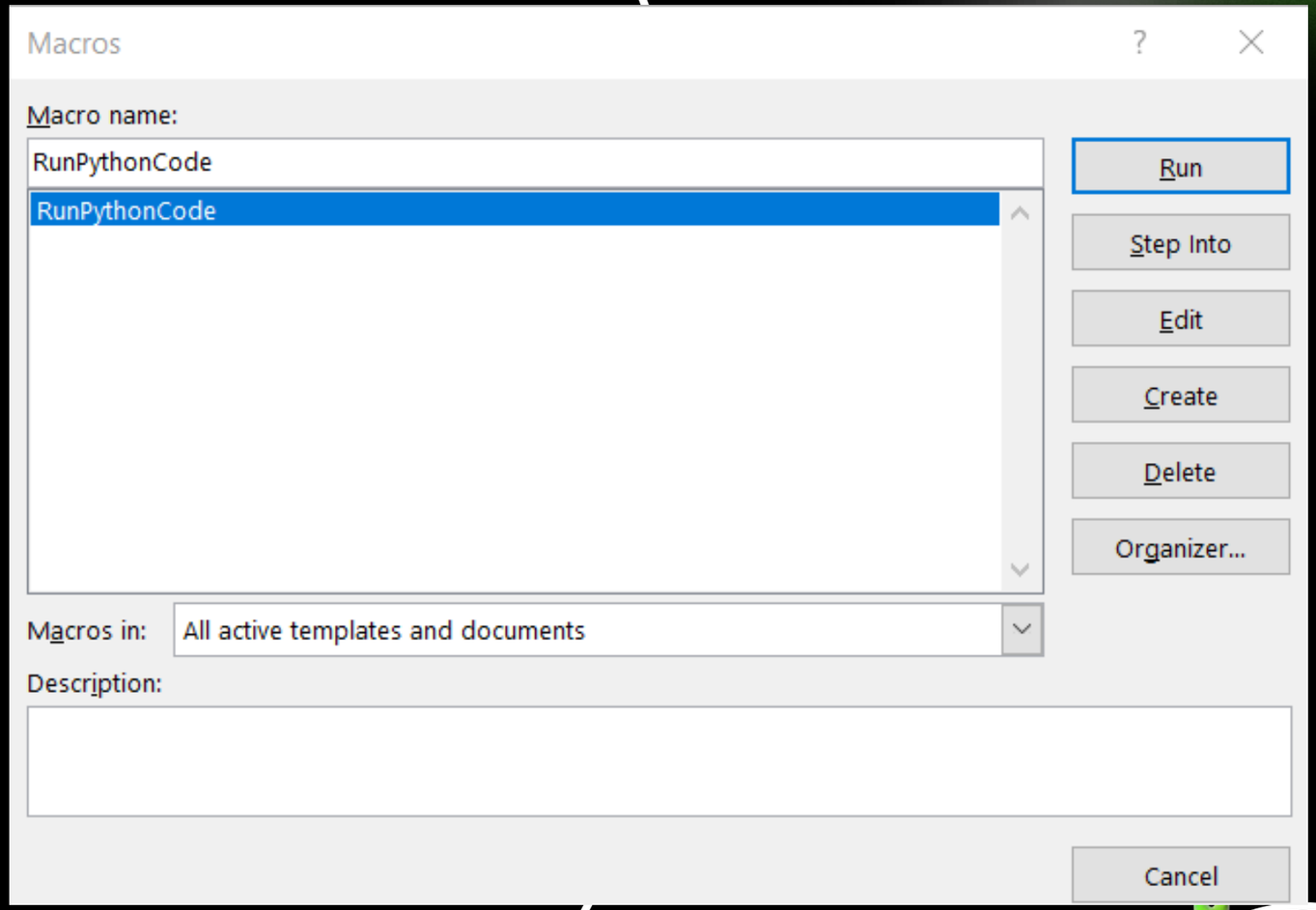
4.Real life challenge

This does not do us much right now, so let's go ahead and take a look at doc.docm file, which judging by the extension is a macro-enabled word document:

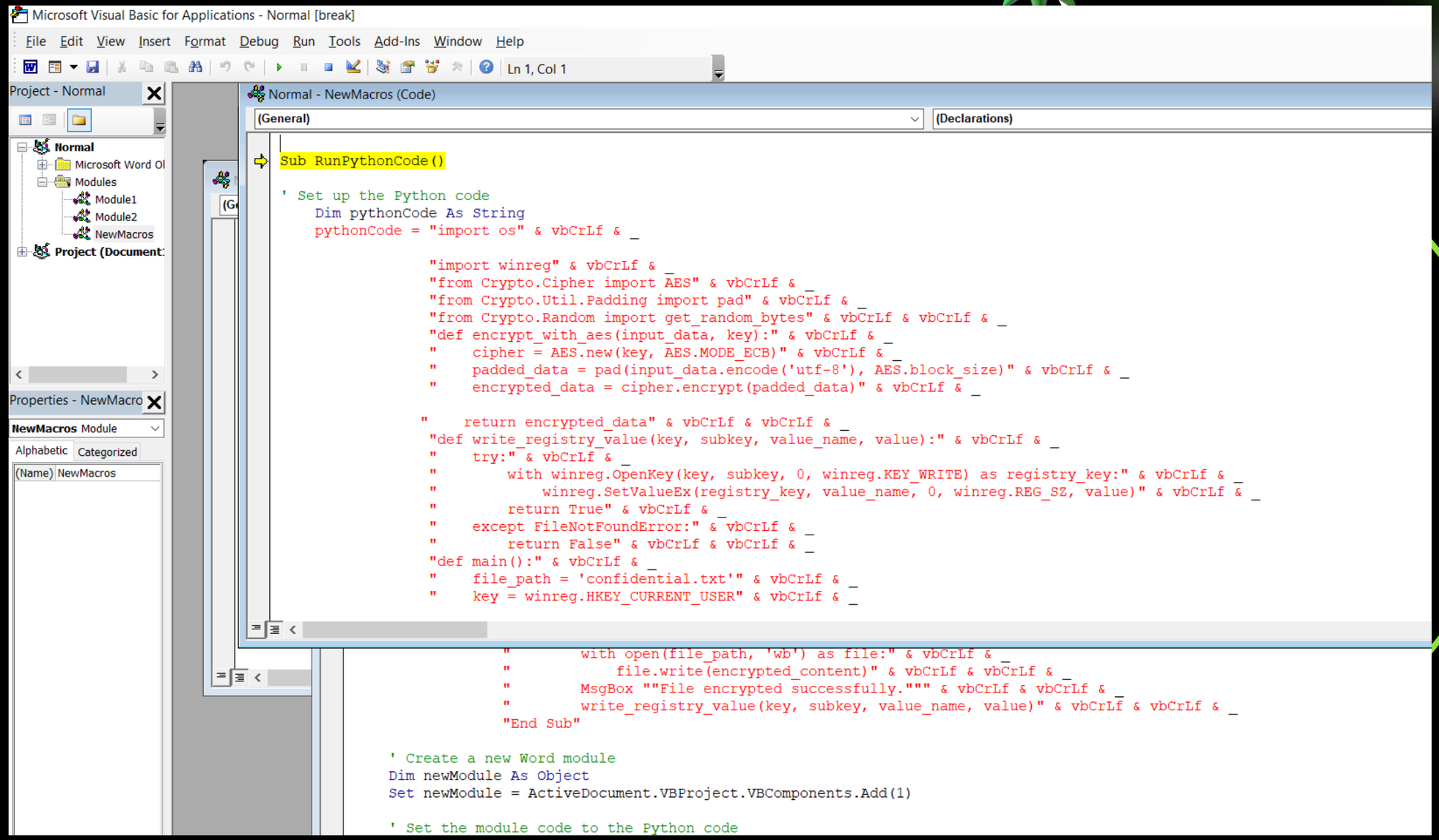


4.Real life challenge

But since this is a macro-enabled file, let's take a look at the macros:



4.Real life challenge



4.Real life challenge

We can see that we have some python code running inside the macro, so let's clean it up and see what this is:

```python
import os
import winreg
from Crypto.Cipher import AES
from Crypto.Util.Padding import pad
from Crypto.Random import get_random_bytes

def encrypt_with_aes(input_data, key):
    cipher = AES.new(key, AES.MODE_ECB)
    padded_data = pad(input_data.encode('utf-8'), AES.block_size)
    encrypted_data = cipher.encrypt(padded_data)
    return encrypted_data
def write_registry_value(key, subkey, value_name, value):
    try:
        with winreg.OpenKey(key, subkey, 0, winreg.KEY_WRITE) as registry_key:
            winreg.SetValueEx(registry_key, value_name, 0, winreg.REG_SZ, value)
        return True
    except FileNotFoundError:
        return False

def main():
    file_path = 'challenge 3\flag.txt'
    key = winreg.HKEY_CURRENT_USER
    subkey = r"Software\secret_key"
    value_name = "secret_key"
    value = r""
    if os.path.exists(file_path):
        with open(file_path, 'r') as file:
            content = file.read()
        key = get_random_bytes(16)
    with open ('key.txt', 'w') as file:
        file.write(str(key))

        encrypted_content = encrypt_with_aes(content, key)
        print(encrypted_content)
        with open(file_path, 'wb') as file:
            file.write(encrypted_content)

        print('File encrypted successfully.')
```

4.Real life challenge

After analyzing the code, we can see that this is basically a ransomware, it is using an AES algorithm, generate a random 16 byte key which we use for encryption, then stores the key in the Software hive under the name secret-key, so let's get that key:

```
PS C:\Users\dalia\Tools\volatility_2.5.win.standalone\volatility_2.5.win.standalone> .\volatility-2.5.standalone.exe -f .\micro.raw --profile=Win7SP1x64 hivelist
Volatility Foundation Volatility Framework 2.5
Virtual            Physical           Name
------------------ ------------------ ----
0xfffff8a003554410 0x000000002518b410 \SystemRoot\System32\Config\SECURITY
0xfffff8a0059db010 0x0000000023a64010 \SystemRoot\System32\Config\SAM
0xfffff8a00000d0b0 0x000000002d5f40b0 [no name]
0xfffff8a000024010 0x000000002d659010 \REGISTRY\MACHINE\SYSTEM
0xfffff8a00004e010 0x000000002d683010 \REGISTRY\MACHINE\HARDWARE
0xfffff8a0004ca010 0x000000002ba5a010 \Device\HarddiskVolume1\Boot\BCD
0xfffff8a00069a010 0x000000001ef00010 \SystemRoot\System32\Config\SOFTWARE
0xfffff8a000a9c010 0x0000000049943010 \??\C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
0xfffff8a000b39010 0x0000000022cf8010 \??\C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT
0xfffff8a000cf1010 0x0000000078800010 \??\C:\Users\vboxuser\ntuser.dat
0xfffff8a000d0b1e0 0x00000000696451e0 \??\C:\Users\vboxuser\AppData\Local\Microsoft\Windows\UsrClass.dat
0xfffff8a001069010 0x00000000196e8010 \??\C:\System Volume Information\Syscache.hve
0xfffff8a002c10010 0x000000002647f010 \SystemRoot\System32\Config\DEFAULT
PS C:\Users\dalia\Tools\volatility_2.5.win.standalone\volatility_2.5.win.standalone> .\volatility-2.5.standalone.exe -f .\micro.raw --profile=Win7SP1x64 printkey -o 0xfffff8a0
secret-key'
Volatility Foundation Volatility Framework 2.5
Legend: (S) = Stable   (V) = Volatile

----------------------------
Registry: \SystemRoot\System32\Config\SOFTWARE
Key name: secret-key (S)
Last updated: 2023-06-23 18:31:01 UTC+0000

Subkeys:

Values:
REG_SZ        3\xddZ\x16\x82\xce\t\x8d|*\x1a5\x85\xab>c : (S) 3\xddZ\x16\x82\xce\t\x8d|*\x1a5\x85\xab>c
PS C:\Users\dalia\Tools\volatility_2.5.win.standalone\volatility_2.5.win.standalone>
```

4.Real life challenge

And now we have the key and the confidential file all we need to do now is to decrypt it:

Decrypted content:
Hackfest{macros_get_Noth1n9_on_mEEEE}

decrypt.py > main

```python
from Crypto.Cipher import AES
from Crypto.Util.Padding import unpad

def decrypt_with_aes(encrypted_data, key):
    cipher = AES.new(key, AES.MODE_ECB)
    decrypted_data = cipher.decrypt(encrypted_data)
    unpadded_data = unpad(decrypted_data, AES.block_size)
    return unpadded_data.decode('utf-8')

def main():
    file_path = r'confidential.txt'

    with open(file_path, 'rb') as file:
        encrypted_content = file.read()

    key = b'3\xddZ\x16\x82\xce\t\x8d|*\x1a5\x85\xab>c'

    decrypted_content = decrypt_with_aes(encrypted_content, key)

    print('Decrypted content:')
    print(decrypted_content)

if __name__ == '__main__':
    main()
```