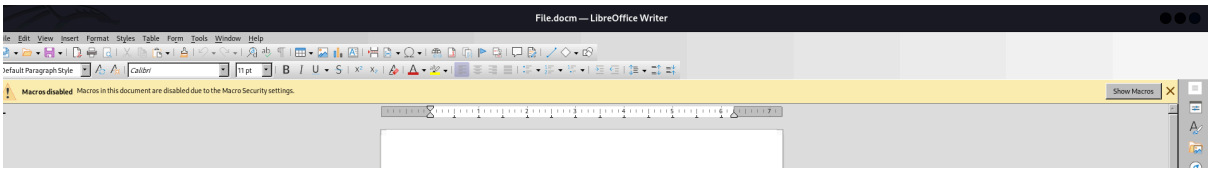
Decrypting the Ransom: Malicious DOCM Analysis

Description : A challenge where the goal was to analyze a malicious DOCM file, extract the encryption key from the ransomware, and decrypt the encrypted data.

Solution:

Opening the File

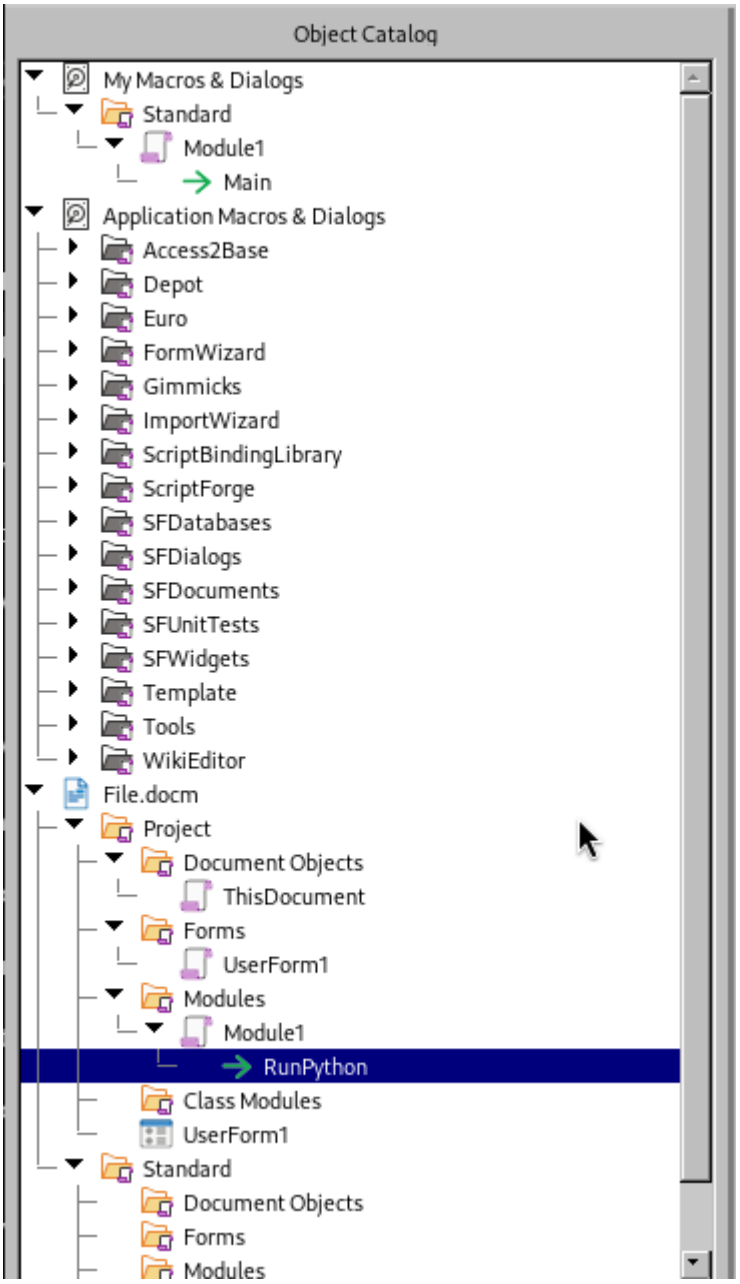
- The challenge provided a `.docm` file, which was opened using **LibreOffice**.
- 

Noticing Macros

- Observed the **"Show Macros"** option in the top menu of LibreOffice. This immediately indicated that the challenge might revolve around embedded macros.

Accessing Macros

- Navigated to **Tools > Macros > Edit Macros** to examine the macro scripts embedded in the document.
- This opened the **Edit Macros** page in LibreOffice.

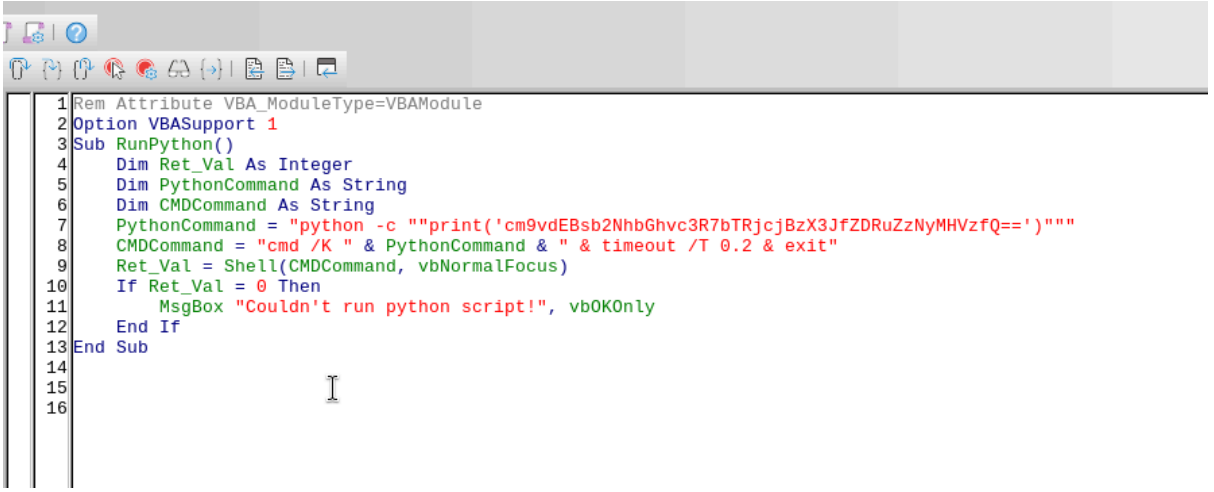
- 

Identifying the Suspicious Macro

- Under the **Modules** tab, found a file named **RunPython**. This script stood out as potentially significant.

Analyzing the Macro

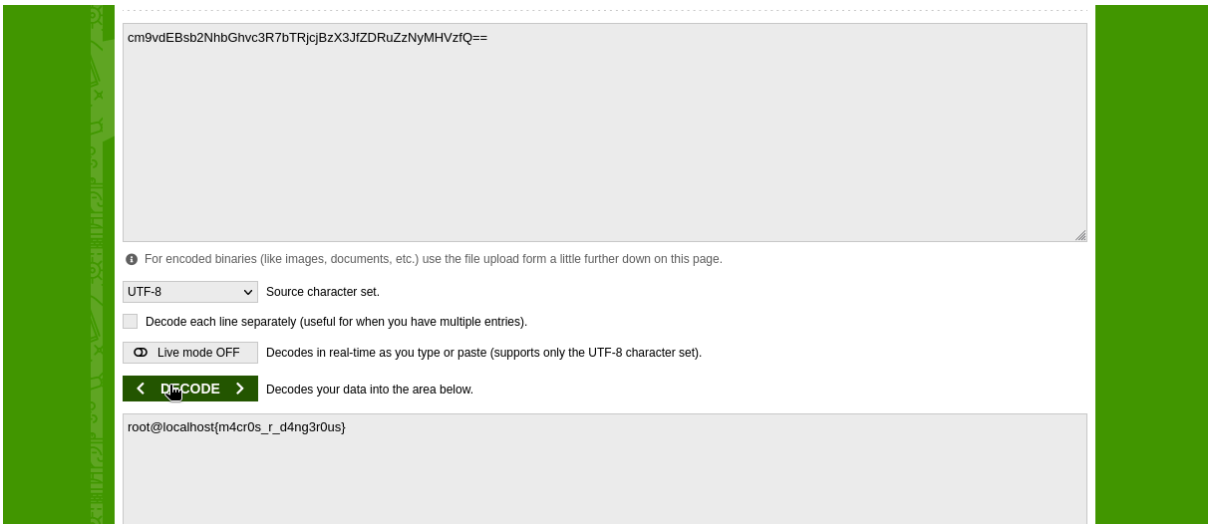
- Opened the `RunPython` script for inspection and began analyzing its code.

- 

```
Rem Attribute VBA_ModuleType=VBAModule
Option VBASupport 1
Sub RunPython()
    Dim Ret_Val As Integer
    Dim PythonCommand As String
    Dim CMDCommand As String
    PythonCommand = "python -c ""print('cm9vdEBsb2NhbGhvc3R7bTRjcjBzX3JfZDRuZzNyMHVzfQ==')"""
    CMDCommand = "cmd /K " & PythonCommand & " & timeout /T 0.2 & exit"
    Ret_Val = Shell(CMDCommand, vbNormalFocus)
    If Ret_Val = 0 Then
        MsgBox "Couldn't run python script!", vbOKOnly
    End If
End Sub
```

-
-
- Found a **Base64 encoded string** embedded within the macro logic. This string seemed like it could be the flag to solving the challenge.

Decoding the Base64 String



Submitting the Flag

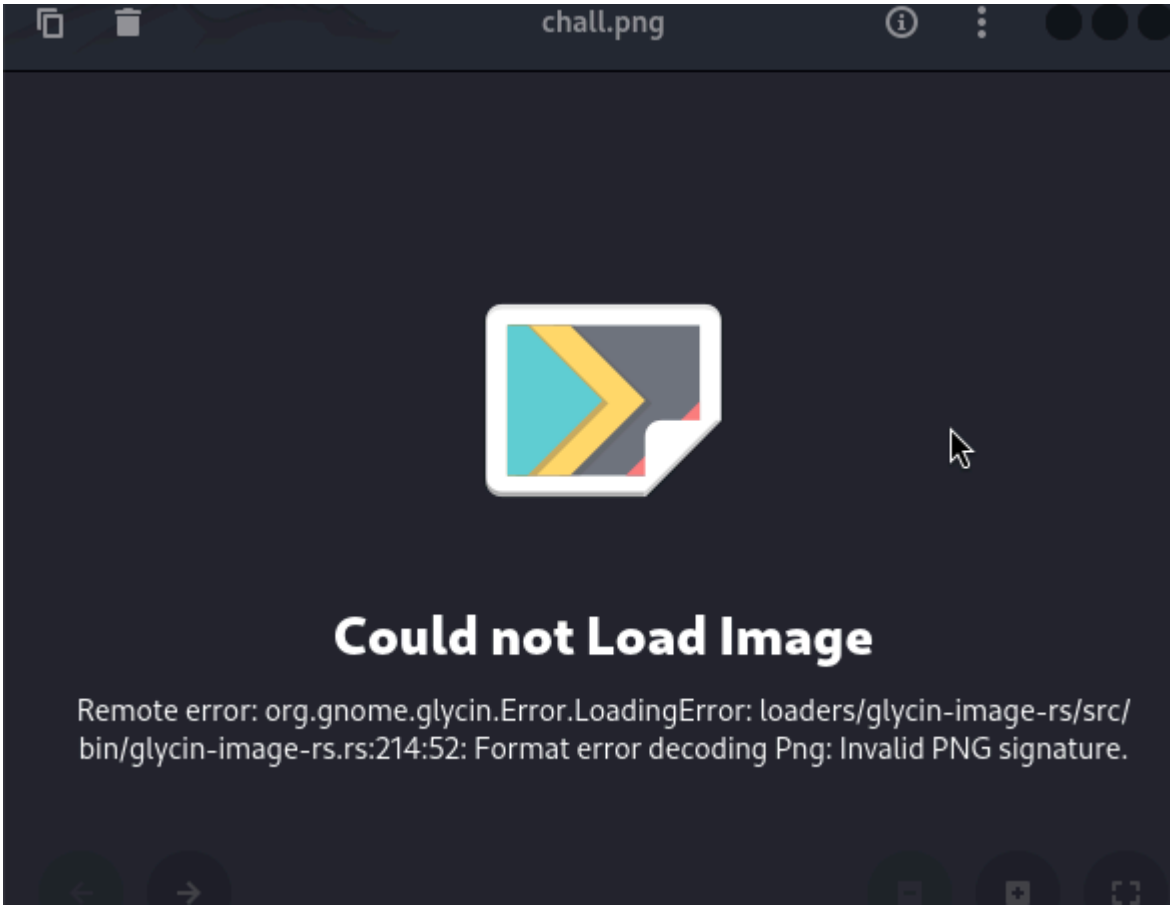
- Used the decoded flag to complete the challenge.

EDIT

Description: In a forgotten data vault, a technician discovers a peculiar file, its contents scrambled and unreadable. There's no obvious way to decode it, but something feels off — as if the file is waiting for the right touch to restore its original form.

Solution:

Inspecting the Provided File

- The challenge provided a **PNG file** for analysis.
- Attempted to open the file using standard image viewers, but it appeared to be **corrupted**.
- 

Researching a Solution

- Searched online for tools or techniques to recover corrupted image files.
- Discovered a GitHub repository named **MagicBytes**:
  - **GitHub Link:** [MagicBytes by Haxrein]
  - This tool specializes in **repairing damaged headers** specific to various image formats, including PNG.

Cloning the Repository

Cloned the GitHub repository to the local machine using:

```
git clone https://github.com/Haxrein/MagicBytes.git
```

Running the Tool

- Installed any necessary dependencies (if prompted) using `pip`.
- Used the tool to recover the corrupted PNG file:

- 

Opening the Recovered Image

- Opened the recovered image using an image viewer.
- 

- The **flag** was displayed within the image.

Submitting the Flag

- Retrieved and submitted the flag to complete the challenge.

fsociety Takeover

Description: Elliot Alderson has left traces of his work while hacking E Corp. Your mission is to uncover the **three hidden keys** on this machine, each representing a step in his plan.

Rules:

1. Find all three keys and document your steps.
2. Include a **timestamp screenshot** of the keys with your machine's local time.
3. Submit your write-up through a Discord ticket in the #support channel.

A flag will be provided upon verification. Good luck, hackers—**society needs you!**

Solution

Initial Steps(I Had no idea on these commands and concepts, just followed chatgpt's assistance)

1. Extracting the `.ova` File

- The provided `.ova` file was extracted to the local system:

```
tar -xvf challenge.ova -C ./robot
```

- The contents of the `.ova` file were successfully extracted into the `./robot` directory.

2. Converting the `.vmdk` File to a Raw File System

- The `.vmdk` file within the extracted content was converted to a raw file system for further analysis using `qemu-img`:

```
qemu-img convert -O raw ./robot/mrrobot.vmdk fs.raw
```

3. Associating the Raw Image with a Loop Device

- The raw file system image (`fs.raw`) was attached to a loop device using `losetup` to access its partitions:

```
sudo losetup -fP fs.raw
```

4. Mounting the Loop Device

- The loop file system was mounted to the local machine at `/mnt/robots`:

```
sudo mount /dev/loopXpY /mnt/robots
```

- This step enabled access to the file system for further enumeration.

Enumeration Steps

1. Navigating to the Mounted File System

- Changed to the directory containing the mounted file system:

```
cd /mnt/robots
```

2. Searching for Files Related to Keys

- Used the `find` command to locate files potentially containing the flag:

```
find ./opt -name "*key*" 2>/dev/null
```

3. Finding the Second Part of the Flag

- Located the file containing the second part of the flag:

```
cat ./home/robot/key-2-of-3.txt
```

4. Attempting to Access `/root` Directory

- Tried to `cd` into `/root`, but encountered a **permission denied** error:

```
cd /root
```

5. Escalating Privileges to Access `/root`

- Used `sudo` to bypass the permission restrictions:

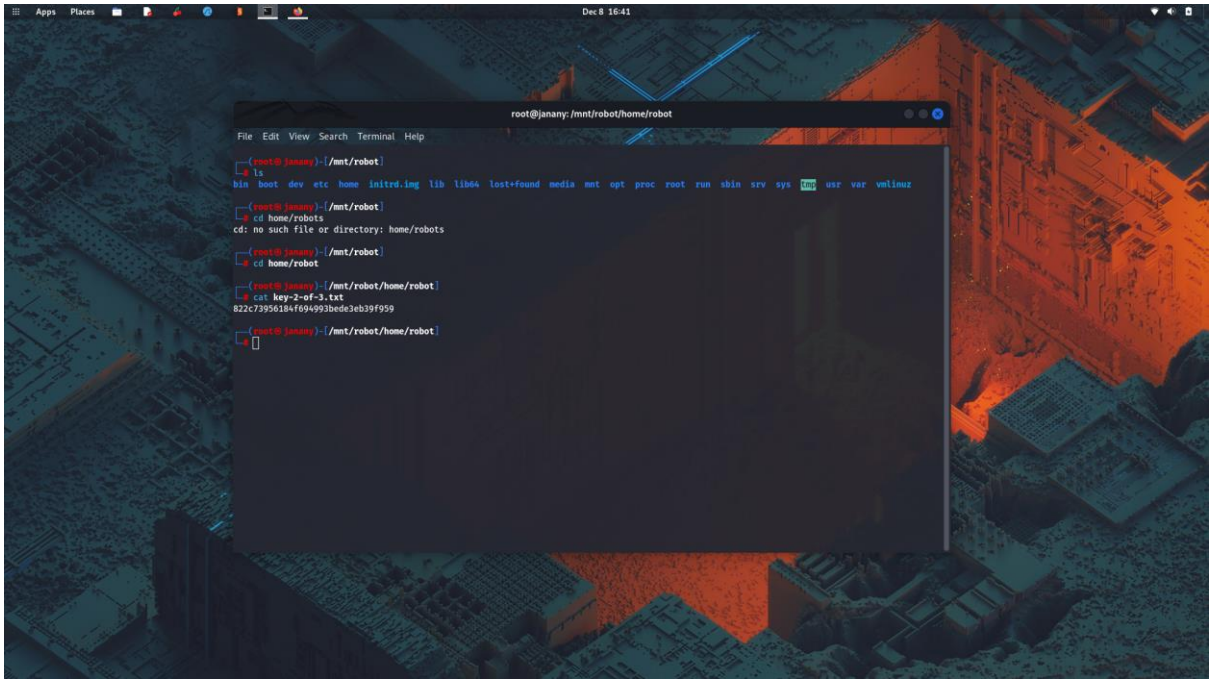
```
sudo cat /root/key-3-of-3.txt
```

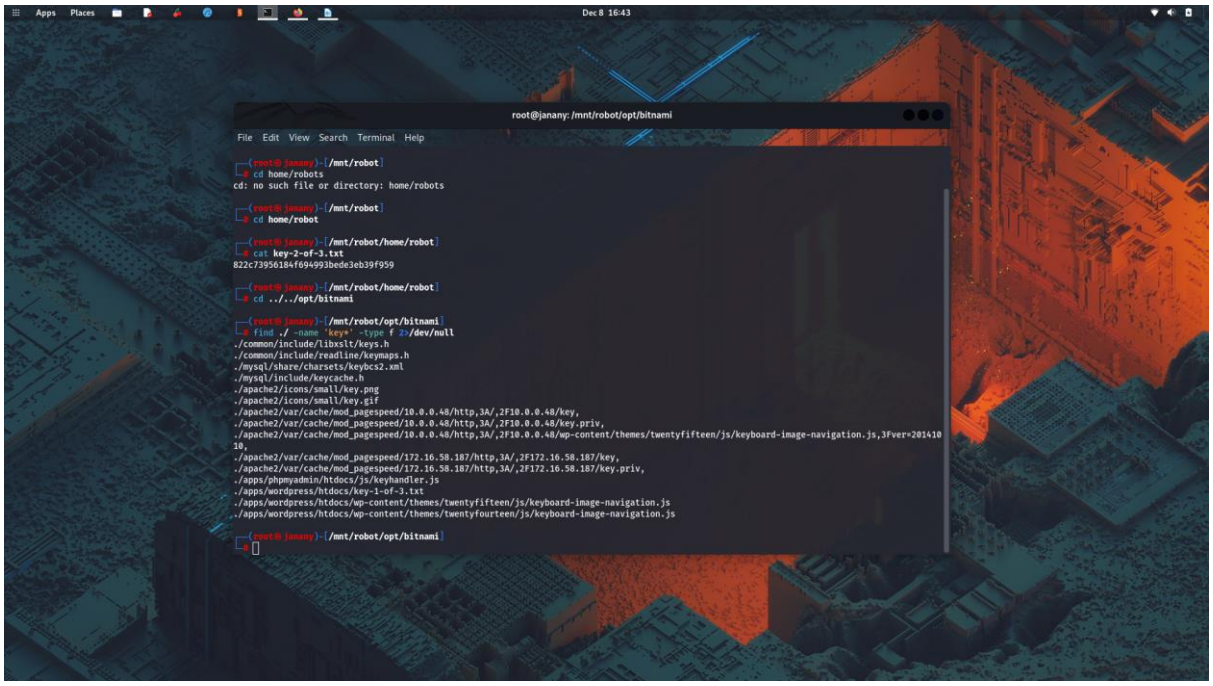
KEYS:

- 073403c8a58a1f80d943455fb30724b9
- 822c73956184f694993bede3eb39f959
- 04787ddef27c3dee1ee161b21670b4e4

Screenshots:

SECOND PART OF KEY:





FIRST PART OF KEY:

```
./apache2/var/cache/mod_pagespeed/10.0.0.48/http,3A/,2F10.0.0.48/wp-content/themes/twenty
10,
./apache2/var/cache/mod_pagespeed/172.16.58.187/http,3A/,2F172.16.58.187/key,
./apache2/var/cache/mod_pagespeed/172.16.58.187/http,3A/,2F172.16.58.187/key.priv,
./apps/phpmyadmin/htdocs/js/keyhandler.js
./apps/wordpress/htdocs/key-1-of-3.txt
./apps/wordpress/htdocs/wp-content/themes/twentyfifteen/js/keyboard-image-navigation.js
./apps/wordpress/htdocs/wp-content/themes/twentyfourteen/js/keyboard-image-navigation.js

┌──(root㉿janany)-[/mnt/robot/opt/bitnami]
└─# cat ./apps/wordpress/htdocs/key-1-of-3.txt

073403c8a58a1f80d943455fb30724b9

┌──(root㉿janany)-[/mnt/robot/opt/bitnami]
└─# z
```

THIRD PART OF KEY:

```
Dec 8 16:45
root@janany: /mnt/robot/root
File  Edit  View  Search  Terminal  Help
./mysql/include/keycache.h
./apache2/icons/small/key.png
./apache2/icons/small/key.gif
./apache2/var/cache/mod_pagespeed/10.0.0.48/http,3A/,2F10.0.0.48/key,
./apache2/var/cache/mod_pagespeed/10.0.0.48/http,3A/,2F10.0.0.48/key.priv,
./apache2/var/cache/mod_pagespeed/10.0.0.48/http,3A/,2F10.0.0.48/wp-content/themes/twentyfifteen/js/keyboard-image-navigation.js,3Fver=201410
10,
./apache2/var/cache/mod_pagespeed/172.16.58.187/http,3A/,2F172.16.58.187/key,
./apache2/var/cache/mod_pagespeed/172.16.58.187/http,3A/,2F172.16.58.187/key.priv,
./apps/phpmyadmin/htdocs/js/keyhandler.js
./apps/wordpress/htdocs/key-1-of-3.txt
./apps/wordpress/htdocs/wp-content/themes/twentyfifteen/js/keyboard-image-navigation.js
./apps/wordpress/htdocs/wp-content/themes/twentyfourteen/js/keyboard-image-navigation.js

┌──(root㉿janany)-[/mnt/robot/opt/bitnami]
└─# cat ./apps/wordpress/htdocs/key-1-of-3.txt

073403c8a58a1f80d943455fb30724b9

┌──(root㉿janany)-[/mnt/robot/opt/bitnami]
└─# cd ../root
cd: no such file or directory: ../root

┌──(root㉿janany)-[/mnt/robot/opt/bitnami]
└─# cd ../../root

┌──(root㉿janany)-[/mnt/robot/root]
└─# ls
firstboot_done  key-3-of-3.txt

┌──(root㉿janany)-[/mnt/robot/root]
└─# cat key-3-of-3.txt
04787ddef27c3dee1ee161b21670b4e4

┌──(root㉿janany)-[/mnt/robot/root]
└─#
```